2014

Load-deformation behaviour of rough rock fractures subjected to internal water flow

Chandrasiri Kumara Wanayalage
University of Wollongong

UNIVERSITY OF WOLLONGONG

School of Civil, Mining and Environmental Engineering

LOAD-DEFORMATION BEHAVIOUR OF ROUGH ROCK FRACTURES SUBJECTED TO INTERNAL WATER FLOW

Chandrasiri Kumara Wanayalage

B.Sc. Eng. (Hons)

This thesis is presented as part of the requirements for the award of the Degree of Doctor of Philosophy of the University of Wollongong

May 2014

THESIS CERTIFICATION

I, Chandrasiri Kumara Wanayalage, declare that this thesis, submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the School of Civil, Mining, and Environmental Engineering, Faculty of Engineering, University of Wollongong, is wholly my own work unless otherwise referenced or acknowledged. The document has not been submitted for qualification at any other academic institution.

……..

Chandrasiri Kumara Wanayalage

May 2014

ABSTRACT

Underground rock strata are often fractured and their permeability is mainly governed by interconnected fracture networks. Flow through fractures must be studied in order to design and operate underground activities such as tunnelling and mine operations, as well as groundwater and petroleum extraction. Flow through a fracture is primarily influenced by its aperture, and because fracture apertures can be distributed widely within a rockmass, they have closures as well as wide openings depending on the location and in-situ stress conditions. Past research studies have been carried out on defining the equivalent aperture to predict fracture flows from uni-directional flow models. However, in most civil engineering applications, plane strain conditions can be assumed (e.g. tunnels, rock slopes), and in such situations two-dimensional fracture models have been suggested for stationary fracture walls. Modelling flow through deformable fractures in plane stain, two-dimensional domain would provide profound insight into rock fracture hydraulics, and these models available now have been simulated using common numerical flow solvers. In this regard, a customised numerical solver to simulate fracture hydraulics would be an important addition to this research area.

In contrast to available literature, in this PhD study, an equivalent two-dimensional flow model was derived from the three-dimensional Navier-Stokes theory for deformable rock fractures. The proposed model contains pressure-velocity coupled equations, and a numerical solution is subsequently introduced by modifying the SIMPLE (Semi Implicit Method for Pressure Linked Equations) algorithm. The Writer's own computer programme (Rock Fracture Flow Solver or RFFS) was developed to solve the proposed model using MATLAB computer language.

Laboratory experiments were carried out for mated and dislocated fracture specimens using the high pressure triaxial apparatus (HPTPTA) designed and built in University of Wollongong. The fracture apertures were measured by replicating them and scanning the surfaces using a 3D laser scanner. Flows through the rock fractures were simulated using the Rock Fracture Flow Solver (RFFS), and the validity of the proposed model was verified for general underground fracture flow situations.

ACKNOWLEDGEMENTS

I wish to express my heartfelt gratitude to everyone who contributed towards the successful completion of my PhD study. First and foremost, I honestly acknowledge my supervisor Prof. Buddhima Indraratna for his support, encouragement and enthusiastic guidance given to me throughout the duration of my research project. It was a huge advantage for me to have his continuous and enormous advises for my work in achieving my goals. I would also like to acknowledge the support and encouragement given by my co-supervisor, Prof. Song-Ping Zhu for my research work. A/Prof. Cholachat Rujikiatkamjorn's help in various times of the research project is greatly acknowledged. I wish to acknowledge the help given by Dr. Jan Nemcik for my laboratory experiments, especially during the time I struggled with lab work.

I would greatly acknowledge the committed efforts of Mr. Alan Grant, who always helped me with technical problems. The support given by all other technical staff including Mr. Ritchie Mclean, Mr. Frank Crabtree, Mr. Ian Laird, Mr. Cameron Neilson and Mr. Fernando Escribano are also acknowledged. Support given by Dr. Winton Gale from Strata Control Technologies Pty Ltd. in collecting data and specimens for laboratory experiments is acknowledged remarkably. I would like to extend my past and present fellow research colleagues of the Centre for Geomechanics and Railway Engineering, University of Wollongong, especially, Dr. Wuditha Premadasa, Dr. Zhenyu Zhang, Mr. Thirukumaran Sivanathan, Dr. Udeshini Pathirage and Mr. Darshana Perera for their enormous friendship and support given to me during this study period.

The financial support given to me by the Endeavour Postgraduate Award programme is greatly appreciated and acknowledged for letting me have a stress free study period. And also the Australian Research Council (ARC) for funding the research project and the University of Wollongong for letting me carryout the research are acknowledged thankfully.

Finally my warm appreciation goes to my wife, Subhani for being with me by my side and supporting me in each and every aspect throughout the study and also my parents, my brother and sister are acknowledged for the encouragement given to me at all times.

LIST OF PUBLICATIONS

Indraratna, B., Kumara, C., Zhu, S., Sloan, S. (2014) 'Mathematical Modelling and Experimental Verification of Fluid Flow through Deformable Rough Rock Fractures'. *International Journal of Geomechanics*, DOI: 10.1061/(ASCE)GM.1943-5622.0000413

Zhang, Z., Indraratna, B., Kumara, C., Nemcik, J. & Gale, W. (2012). Effects of normal stress on water flow through a single rough walled rock joint. In G. A. Narsilio, A. Arulrajah & J. Kodikara (Eds.), *11th Australia - New Zealand Conference on Geomechanics: Ground Engineering in a Changing World* (pp. 113-118)

LIST OF ABBREVIATIONS AND SYMBOLS

A	Cross sectional area
$\overline{\boldsymbol{F}}$	Body force
F_l / F_t	Fracture lower / top wall profile in 3D
$\mathrm{F_T(x,y)_0}$ / $\mathrm{F_B(x,y)_0}$	Initial surface profiles of the top / bottom joint walls
HPTPTA	High pressure two-phase triaxial apparatus
$\overline{\overline{\mathbf{I}}}$	Unit tensor
I / J	Scalar grid indes in X-direction / Y-direction
JRC	Joint roughness coefficient
JCS	Joint compressive strength
K	Hydraulic conductivity
$K_i,\ k_i$	Initial joint normal stiffness
MG, P, N, SE	Specimen names
N_p	Number of discrete points in the x-direction
P	Total pressure
$\overline{\overline{\mathbf{T}}}$	Stress tensor
$\mathbf{V}$	Velocity vector
$\overline{\boldsymbol{V}}$	Depth-averaged velocity
$\widetilde{\boldsymbol{V}}$	Difference between the velocity and depth-averaged velocity
$X^{/}, Y^{/}, Z^{/}$	Global coordinate system
d_m	Mean apparent aperture
e	Aperture
$\bar{\boldsymbol{e}}_{\boldsymbol{z}}$	Unit vector in the vertical direction
$f(e)$	Normalised aperture frequency distribution

f_l / f_t	Fracture lower / top wall profile in 2D
g	Gravitational acceleration
i / j	Staggered grid indes in X-direction / Y-direction
k, $\mathrm{k_{cr}}$	Intrinsic permeability
k_n	Joint normal stiffness
k_s	Joint shear stiffness
⟨k⟩	Flow conductance
nx / ny	Number of finite volumes in X-direction / Y-direction
p	Fluid pressure
r	Relaxation factor
u_w	Pore pressure
v_m	Maximum closure of the joint
x, y, z	Discontinuity oriented coordinate system
z_2	Root mean square of the first derivative of a surface profile
z_l / z_t	z coordinates of the fracture lower / top wall profiles
α	Ratio of the contact area to the total area of the fracture
τ	Shear stress acting on the fracture
λ	Wavelength of aperture variation
σ_n /$\sigma_n^{/}$	Normal stress / effective normal stress acting on the fracture
ρ	Fluid density
μ	Dynamic viscosity
δ_m	Total deformation of the membrane
Δx /Δy	Grid size in X-direction / Y-direction
Δv or δ_n	Vertical or normal deformation
Δd_h	Horizontal deformati

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1 Introduction

1.1 Background of the Research

The permeability of rock masses is a vital parameter in underground design and construction activities, and must be calculated to an acceptable accuracy in underground mines and other projects that are being developed. Although there are highly porous rocks, the matrix permeability of fractured rocks can mostly be omitted, unlike the contribution made by prominent fractures. The interconnected network of fractures permeates fluid (groundwater) in rock masses, so being able to determine the permeability of a single fracture means that the total permeability of the rock mass can then be computed numerically; this is why modelling the characteristics of single fracture flow becomes pertinent in rock fracture hydraulics. Using mathematical models, the determination of flow through rock fractures has been suggested, and in particular, one dimensional (uni-directional) flow models have been initially proposed (e.g. Baker 1955; Snow 1968; Gale 1977) and then modified by considering additional controlling parameters to achieve better predictions (Jones 1975; Gangi 1978; Witherspoon *et al.* 1980; Tsang 1984). Two-dimensional models were also developed to account for the aperture roughness as

well as undulating flow behaviour (Amadei & Illangasekare 1992; Indraratna *et al*. 2002; Price and Indraratna 2005; Koyama 2007; Zimmerman and Yeo 2013). One-dimensional flow models use a single value for the aperture, so it can be used to predict the flow rates at different normal stresses with deforming fracture walls. Two-dimensional flow models to simulate rock fracture flows by combining the normal deformation of a fracture would provide profound insight into rock fracture hydraulics. This is important because rock fractures usually deform according to changes in the applied stress and at different depths, whereas the deformation of fractures is dissimilar. Therefore, modelling two-dimensional flow to account for the roughness of fractures together with their normal deformation is an essential and insightful research area in rock joint hydraulics.

The creation of an underground opening alters the hydraulic gradients of rock

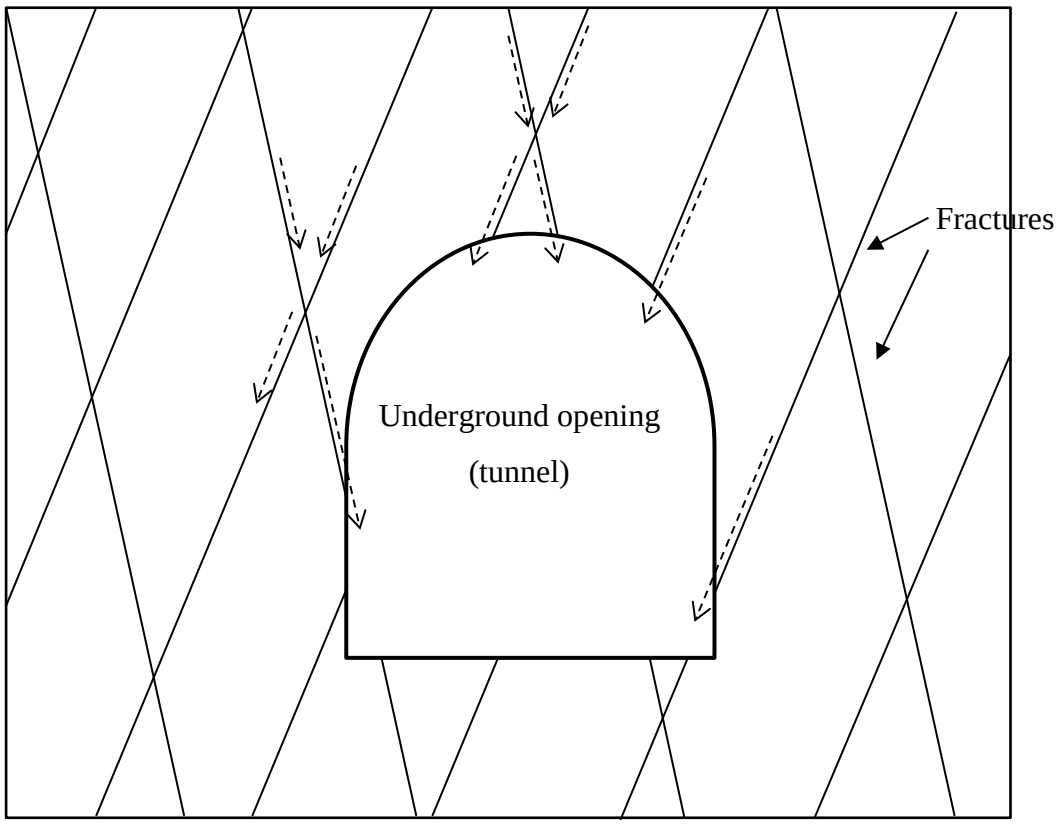


Figure 1.1: Flow towards an underground opening created on a fractured rock.

fractures such that the fracture networks tend to permeate water towards the opening (Figure 1.1). Therefore, in an active underground mine, the directions of fracture flows can change from time to time. Fracture permeability usually depends on the direction of flow that is due to rough surface topography, therefore, the directional permeabilities of fractures is also an important parameter in rock fracture hydraulics. Altering the boundary conditions of fluid pressure, changing the flow direction can be achieved with an appropriate two-dimensional flow model. Development of a two-dimensional flow model for deforming rough rock joints can also be used to calculate directional permeabilities by changing the fluid pressure boundary conditions.

Two-dimensional fracture flow models have been suggested by considering the 2D features from the side of a fracture (Figure 1.2) where the height of the aperture is in one dimension (Indraratna et al. 2002; Price and Indraratna 2005; Koyama 2007; Zimmerman and Yeo 2013). Also there are more advanced models which consider the 2D features of the plan view of a fracture (Bear et al. 1993; Kishida et al. 2013). The former method describes the aperture variation in one direction for unidirectional flows, while the latter describes two-directional flow by taking aperture variation as a variable in 2D space. This method is better at simulating rock fracture flows where the effect of aperture contacts can be included and modelled, and tortuous fluid flow paths can be simulated. However, since the few such models available do not consider the deformation of apertures when applying them in practice, it is difficult to predict the deformation and manually entering them is very challenging and often infeasible. Therefore, developing a model that incorporates

fracture deformations as well as the effect of aperture distribution to the flow is an essential need in rock fracture flow modelling.

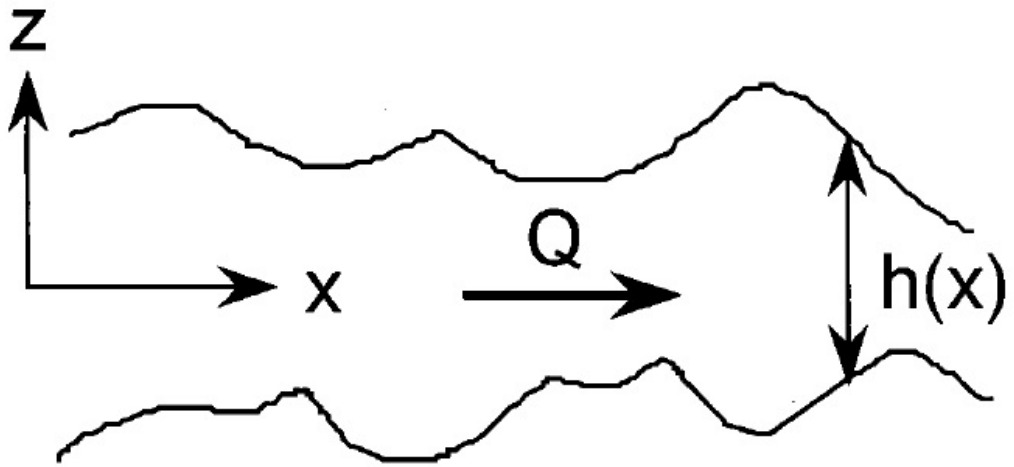


Figure 1.2: 2D fracture model from side view after Zimmerman and Yeo (2013)

Numerical models that can solve partial differential equations are available for many applications and they are essential tools for engineering solutions, but having an exact tailor made numerical model to use when solving a complex mathematical model is not always possible. In these situations, a custom made numerical solver should be developed. Navier-Stokes equations can describe the flow of any kind of fluid or gas, or mixtures of them, but they cannot be solved explicitly, and the existence of a complete solution of full Navier-Stokes equations has not yet been proven. Usually, simplified Navier-Stokes equations are solved numerically according to the application by applying the relevant boundary conditions. Fluid flow through rock fractures can be simulated by creating the fracture mesh in three-dimensional space and then using an available flow solver. For example, Pruess and Tsang (1990) used a general purpose simulator MULKOM to simulate two phase

flow through rock fractures. Zimmerman et al (2004) simulated flow through rock fractures using the flow code FLUIDITY developed by de Oliveira (1999). Koyama (2007), Koyama *et al.* (2008) used the COMSOL finite element solver to simulate fracture flows. Generating the flow domain mesh in 3D is a complex task, and when it comes to analysis it will be impractical to generate each and every flow domain because the undulating fracture walls. A good solution for this problem is the use of 2D models that take the length and width as the dimensions, while the aperture appears in the model as a variable in 2D space. Then there is no need to generate meshes that are cumbersome when the measured aperture heights can be used directly. Kishida et al. 2013 used the highly simplified marker and cell (HSMAC) method to solve the 2D fracture flow model developed for stationery rock fractures. When a 2D flow model has been derived for deformable rock fractures it will be very useful in fracture flow simulations.

The challenge of solving equations derived from Navier-Stokes equations stems from the pressure-velocity coupling. The SIMPLE (semi implicit method for pressure linked equations) developed by Patankar and Spalding (1972) and its later modifications are widely used in commercial solvers for Navier-Stokes equations, but this algorithm has not been extended for 2D fracture flow models where the aperture has been introduced as a new variable. In this instance the SIMPLE algorithm should be modified to cater for the role and deformation of the aperture.

1.2 Objectives of the Research

This research work consists of mathematical modelling, numerical solutions, and experimental verification. The main objectives of the research can be listed as follows

- Comprehensive review of past literature about rock fracture flow modelling, experimental investigations, modelling normal rock fracture deformation, hydro-mechanical coupling of rock fractures, numerical models and simulations of rock fracture flows.
- Development of a new two-dimensional flow model to define the flow of rough rock fractures coupled with the normal deformation of rock fractures under applied normal stress.
- Formulate a numerical solution for the two-dimensional deformation-flow model to couple the pressure and velocity, in order to calculate the distribution of velocity and pressure in the flow domain.
- Develop a computer programme to run the numerical solution in a user friendly environment. Measure the permeability of the rock fracture in longitudinal and transverse directions to calculate the directional permeability of the rock fracture.
- Treat contacts of the rock fracture as local boundaries in the flow domain in the numerical programme to enhance the simulation capability of the numerical model.
- Investigate the flow behaviour of rock fractures experimentally in the laboratory, having conducted permeability tests for real rock fractures, and then verify the developed mathematical model using the experimental parameters and the appropriate computer programme.
- Determine the fracture aperture distribution experimentally in the laboratory so it can be used in the numerical solution, and provide

practical methods to determine the apertures and calculate fracture permeability.

1.3 Organisation of the Thesis

This doctoral thesis consists of seven chapters, including this Introduction, followed by a list of references and appendices.

Chapter 2 provides a comprehensive literature review related to the study area. The initial methods of modelling rock fracture flow, including their improvements and the new models currently available, are described. Next, the numerical solution methods used to solve non-linear models, and additional notes on the method used in this study are elaborated. The chapter finishes with an explanation of the laboratory experiments conducted by past researchers and their findings, together with model verifications.

Chapter 3 contains the development of the writer's own mathematical model and the associated numerical solution for single phase flow in a rough rock fracture. The development of the model is expounded with essential steps and explanations. The numerical model solution using the finite volume method and SIMPLE algorithm is also explained.

In Chapter 4 the development of the computer programme is explained. The programme was developed in the MATLAB design environment using its own computer language to solve the descretised governing equations. Making end user interactions user friendly through graphical user interfaces and additional options to help the user obtain better results are explained in this chapter. The relevant MATLAB programme codes are listed in Appendix A, and are also referred to in Chapter 5.

Chapter 5 describes the laboratory experiment procedures. The first part explains the triaxial apparatus tests performed to calculate the volumetric flow rates through real rock fractures under increasing confining pressure and varying inlet water pressure, followed by a comprehensive description of non-contact laser scanner procedures used to obtain the fracture aperture distribution.

Chapter 6 contains the laboratory results and model verification, based on the experimental study. Using the developed computer programme to study flow characteristics of rock fractures is also discussed in Chapter 6.

Chapter 7 provides the Conclusion, including the research outcomes and their uses, the limitations of the developed models, and recommendations for future research in this area.

Chapter 2 Literature Review

2.1 Rocks and Rock Fractures

Rocks exist in the earth's crust as a cycle with a great chronological span. There are major three types of rocks based on their origin namely, sedimentary, metamorphic, and igneous, and after they have been formed by any of these methods they experience external stresses which when prolonged, cause them to fracture. As well as fractures there are other discontinuities such as faults, folds, and bedding planes. Fractures or joints are the most common discontinuities, and any rocks in the earth's crust will have at least fractures in a micro scale. Initialised micro fractures will grow into wider fractures if they suffer from continuous stresses. Fractures in rocks affect construction activities in many ways because fractures reduce the strength of rock mass. In the mining industry many remedial measures are used to strengthen fractured rock, such as rock bolting and shotcreating whereas in the petroleum industry, fractures provide storage and transport for petroleum resources thereby supporting production.

Interconnectivity of fractures in a rock provides flow paths, and in the mining industry the flow through rock fractures plays a major role. The creation of

underground openings generates hydraulic gradients that induce fracture flows that affect underground activities and cause inundation if accumulated groundwater is not removed efficiently. In petroleum extraction processes, hydraulic gradients are created artificially inducing fracture flows. In all these cases, understanding fracture flow is a significant requisite that leads to the study of rock fracture flows.

2.2 Importance of Rock Hydraulics

Underground operations such as mining, petroleum extraction, tunnels, and nuclear waste storage, etc. often involve jointed rockmass. Rocks are fractured in different magnitudes and ranges according to the stress environment. The matrix permeability of intact rock is extremely low because the grains are closely packed, and therefore the permeability of an intact (non-fractured) rock is negligible. However, fractured rocks have considerable permeability when the fractures became interconnected. Geotechnical activities are affected in numerous ways when the permeability rises due to interconnected fracture networks. Underground mines are usually well below the water table and that provides hydraulic gradients towards the mine openings which cause water ingress. Mines affected in such ways should be designed to remove the groundwater to stop inundation to sustain smooth mining operations.

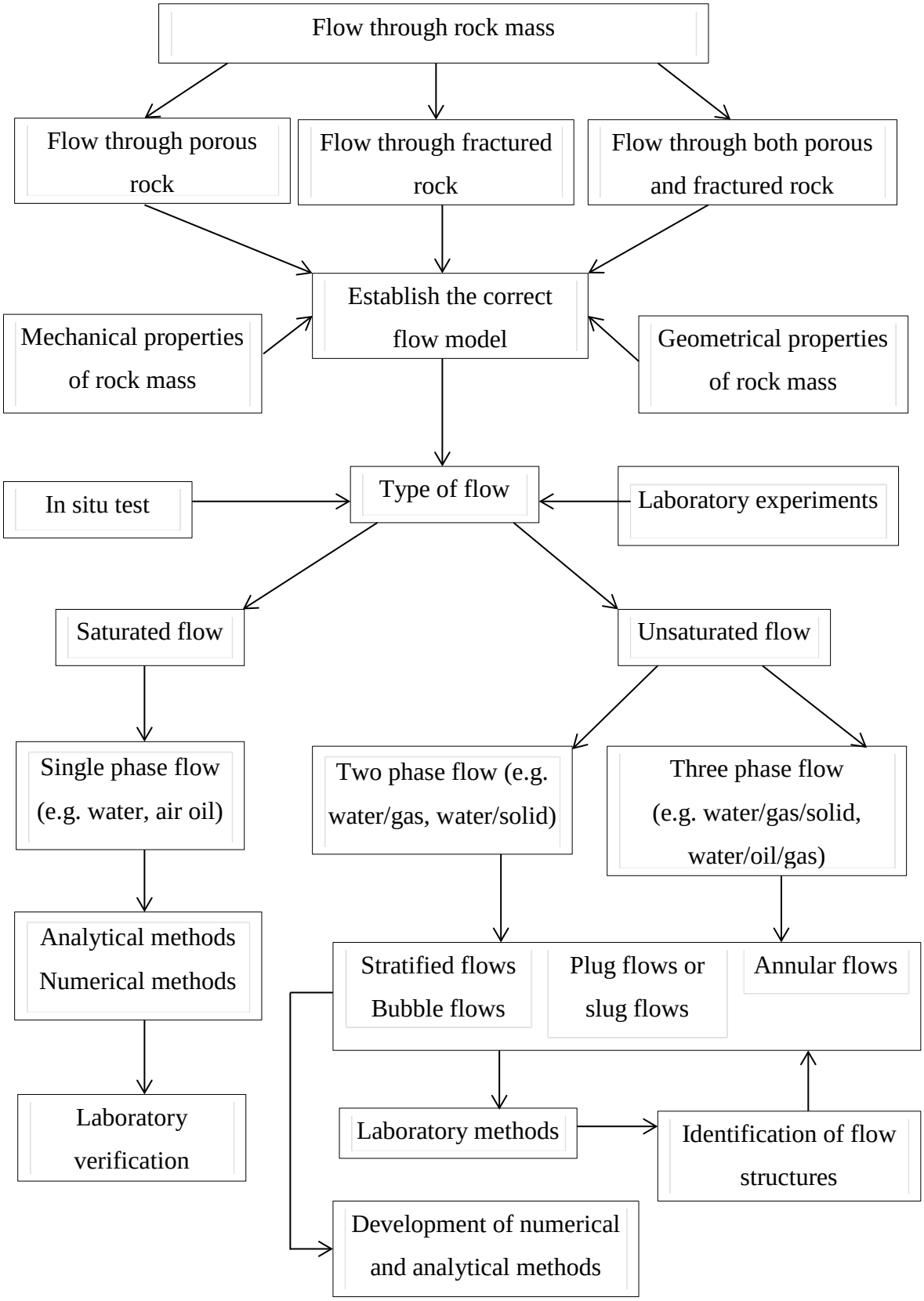


Figure 2.1: Flow chart for flow analysis through a rock mass after Indraratna *et al.* (1999)

Fractures in rock not only reduce its strength, but also their presence can make mining difficult, in contrast to petroleum industry where fractured rocks are often beneficial. Fractured rocks act as reservoirs for liquid petroleum as well as transporting liquid ore towards the production wells. Therefore, a sound knowledge of the parameters that affect rockmass permeability and its behaviour will result in better planning and maintenance.

Although the rock matrix has intrinsic permeability through its grain structure, fracture permeability dominates the total permeability of rocks when the matrix permeability is very low. In these cases, the permeability of the rock matrix often is omitted, but for relatively porous rocks, the dual permeability should be considered. Hydro-Mechanical coupled advanced constitutive models have been developed for macroscopic porous media (Khalili et al. 2008) and double porous media (Khalili and Selvadurai 2003) to describe flow and deformation behaviour of saturated and unsaturated soils. Gens *et al*. (2008) and Garitte *et al.* (2014) developed Thermo-Hydro-Mechanical (THM) and Hydro-Mechanical formulation and constitutive models to analyse coupled problems involves with rocks such as excavations. Pak and Chan (2004) also modelled fully implicit THM fracture model for hydro fracturing in petroleum industry. Indraratna *et al.* (1999) designed a flow chart that analysed flow through rock mass and addressed all the possibilities including the type of flow (Figure 1.1).

2.3 One-Dimensional Rock Fracture Flow Models

Predicting the flow of fluid through a rock fracture is essential when dealing with fractured rocks. A rock fracture can be assumed to be a duct (Figure 2.2) with a very small height compared to its length and breadth. The usual theories applicable to

rectangular cross sectional pipe flows can be applied as the basis of this assumption, and they can predict the volumetric flow rate through a rock fracture.

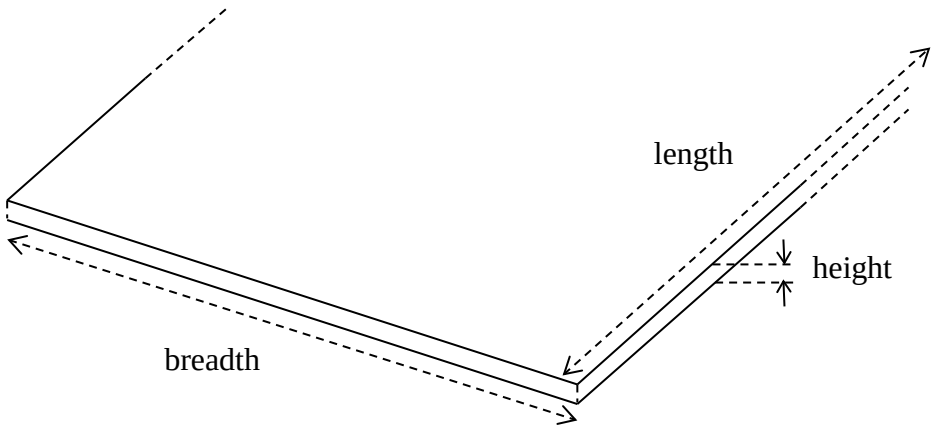


Figure 2.2: A fracture assumed as a duct

The parallel plate model can be assumed as the first simplified model to predict flow through rock fractures. This model considers the two fracture walls to be two parallel plates that remain apart by a distance equal to the aperture (Baker 1955; Snow 1968; Gale 1977). The development of the model from the Navier-Stokes equations explained by Zimmerman and Bodvarsson (1996) can be expressed as follows. The Navier-Stokes equation for the momentum conservation of a Newtonian fluid for divergence free flow is given by Equation (2.1).

$$\frac{\partial \overline{\boldsymbol{u}}}{\partial t} + (\overline{\boldsymbol{u}}.\boldsymbol{\nabla})\overline{\boldsymbol{u}} = \overline{\boldsymbol{F}} - \frac{1}{\rho}\boldsymbol{\nabla} p + \frac{\mu}{\rho}\nabla^{\mathbf{2}}\overline{\boldsymbol{u}} \tag{2.1}$$

In the above $\overline{\boldsymbol{u}}$ is the velocity vector. The first term on the left represents the acceleration of a fluid particle, while the second term on the left is the advective acceleration of a particle due to a change in velocity when a particle moves to a different position with another velocity, even in steady state flow. $\overline{\boldsymbol{F}}$ is the body force, and for subsurface flow, gravity ($-g\bar{\boldsymbol{e}}_{\boldsymbol{z}}$, $\bar{\boldsymbol{e}}_{\boldsymbol{z}}$ is the unit vector in the vertical

direction) will be the only existing body force. The second term on the right is the applied pressure gradient and the third represents viscous forces. The gravitational term was removed from the equation by introducing a reduced pressure, according to Batchelor (1967) such that;

$$P = p + \rho g z \tag{2.2}$$

The pressure due to gravity exists in the same body of fluid at rest and in a moving fluid. Substitution of Equation (2.2) in (2.1) gives Equation (2.3);

$$-g\overline{\boldsymbol{e_z}} - \frac{1}{\rho}\boldsymbol{\nabla} p = \frac{-1}{\rho}\boldsymbol{\nabla}(\rho g z + p) = \frac{-1}{\rho}\boldsymbol{\nabla} P \tag{2.3}$$

Fracture permeability is generally considered by assuming a steady-state flow where the acceleration term is cancelled out and Equation (2.3) reduces to Equation (2.4).

$$\rho(\overline{\boldsymbol{u}}.\boldsymbol{\nabla})\overline{\boldsymbol{u}} = -\boldsymbol{\nabla}\mathrm{P} + \mu\nabla^{\mathbf{2}}\overline{\boldsymbol{u}} \tag{2.4}$$

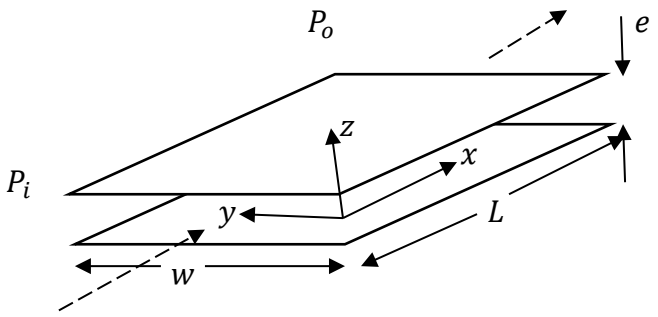


Figure 2.3: Parallel plate model

Figure 2.3 shows the parallel plate model for uni-directional rock fracture flow with an aperture e, a fracture width w and a segment of length L. Water flows in the x direction and the velocity only changes in the z direction for steady flows. The advection term can be expanded as given below.

$$(\bar{\boldsymbol{u}}.\boldsymbol{\nabla})\bar{\boldsymbol{u}} = (\bar{\boldsymbol{u}}.\boldsymbol{\nabla})\big(u_x, u_y, u_z\big) = \big[\bar{\boldsymbol{u}}.(\boldsymbol{\nabla}u_x),\ \bar{\boldsymbol{u}}.\big(\boldsymbol{\nabla}u_y\big), \bar{\boldsymbol{u}}.(\boldsymbol{\nabla}u_z)\big] \tag{2.5}$$

$\boldsymbol{\nabla}u_x$ and $\boldsymbol{\nabla}u_y$ become zero because the velocity gradient only exists in the z direction. Since the velocity vector is in the x direction and $\boldsymbol{\nabla}u_z$ is perpendicular to that, the dot product $\bar{\boldsymbol{u}}.(\boldsymbol{\nabla}u_z)$ becomes zero and the advection term vanishes.

Then Equation (2.4) reduces to Equation (2.6).

$$\boldsymbol{\nabla}\mathrm{P} = \mu\nabla^{\mathbf{2}}\bar{\boldsymbol{u}}(z) \tag{2.6}$$

The velocity vector is written as $\bar{\boldsymbol{u}}(z)$, hence *z* is the only variable. The pressure gradient is in the *x* direction, therefore the pressure gradient $\boldsymbol{\nabla}\mathrm{P}$ can be written as follows.

$$\boldsymbol{\nabla}\mathrm{P} = \left(\frac{\partial \mathrm{P}}{\partial \mathrm{x}}, \frac{\partial \mathrm{P}}{\partial \mathrm{y}}, \frac{\partial \mathrm{P}}{\partial \mathrm{z}}\right) = \left(\frac{\partial \mathrm{P}}{\partial \mathrm{x}}, 0, 0\right) \tag{2.7}$$

Then Equation (2.6) can be expressed in scalar form as follows:

$$\nabla^{\mathbf{2}}u_{\boldsymbol{x}}(z) = \frac{1}{\mu}\frac{\partial \mathrm{P}}{\partial \mathrm{x}} \tag{2.8}$$

$$\nabla^{\mathbf{2}}u_{\boldsymbol{y}}(z) = 0 \tag{2.9}$$

$$\nabla^{\mathbf{2}}u_{\boldsymbol{z}}(z) = 0 \tag{2.10}$$

Here Equations (2.9) and (2.10), and the boundary conditions of the system satisfy $u_{\boldsymbol{y}}(z) = 0$ and $u_{\boldsymbol{z}}(z) = 0$ respectively. Integrating Equation (2.8) with respect to *z* twice will result in Equation (2.11), thus,

$$u_x(z) = \frac{1}{\mu}\frac{\partial \mathrm{P}}{\partial \mathrm{x}}.\frac{\mathrm{z}^2}{2} + \mathrm{C} \tag{2.11}$$

Applying the boundary condition $u_x(z) = 0\ at\ z = \mp {}^{e}/_{2}$, the constant of integration can be substituted as follows.

$$u_x(z) = \frac{1}{2\mu}\frac{\partial \mathrm{P}}{\partial \mathrm{x}}\left\{\mathrm{z}^2 - \left({}^{\mathrm{e}}/_{2}\right)^2\right\} \tag{2.12}$$

This relationship gives the velocity in the direction of flow. By integrating the velocity across the fracture from $-e/2$ to $+e/2$, the volumetric flux Q_x for fracture width w can be obtained as follows:

$$Q_x = w \int_{-e/2}^{+e/2} \bar{\boldsymbol{u}}_{\boldsymbol{x}}(z)\, dz \tag{2.13}$$

The above can be integrated to obtain Equation (2.14):

$$Q_x = -w\frac{\mathrm{e}^3}{12\mu}\left(\frac{\partial \mathrm{P}}{\partial \mathrm{x}}\right) \tag{2.14}$$

Equation (2.14) is known as the cubic formula because the volumetric flow rate is proportional to the cube of the aperture. By comparing the cubic formula with Darcy's law and taking the cross sectional area $A = ew$, the hydraulic conductivity (K) for fracture flow can be defined as follows.

$$K = \frac{1}{\mu}.\frac{e^2}{12} \tag{2.15}$$

Here $e^2/12$ is called the intrinsic permeability of the fracture and it is a property of the fracture while the hydraulic conductivity is a property of a fluid that permeates through the particular fracture. Usually the intrinsic permeability is denoted by *'k'* while the hydraulic conductivity of the fracture is given by *'K'*.

2.3.1 *Modifications to the Cubic Formula*

The validity of the cubic formula was experimentally verified by several researchers (Lomize 1951; Witherspoon et al. 1980). Since the cubic formula assumes the fracture walls to be plates the flow rates were overpredicted compared to real fracture flows even though an average aperture was assumed. Although there are other factors that affect flow through rock fractures, this relationship is widely used in practice because of its simplicity. The factors that affect flow through fractures are,

- Fracture surface roughness.
- Compressibility of flowing fluids.
- Solubility of flowing fluids.
- Stress environment
- Mechanical deformation of the joint wall.
- Phase changes of flowing fluids.

Although the actual fracture surfaces are always irregular in shape, the surfaces sometimes make contact. As a result there were doubts about using the smooth parallel plate model, so researches were compelled to account for the effect of surface roughness. Iwai (1976) suggested that rough walled fractures obeyed the cubic law when they are subjected to low normal stresses. To study the effect of surface roughness, a fractal model was presented by Patir and Cheng (1978) for their study of the hydrodynamic lubrication of rough bearings, while Brown (1987) extended their results to fluid flow through rock joints.

Gangi (1978) used a bed of nails model for the asperities of the fracture, such that the functional dependence of the variability in the permeability of a fracture with confining pressure was shown as;

$$k(P) = k_0[1 - (P/P_1)^m]^3 \quad (2.16)$$

Where k_0 is zero pressure permeability, P_1 is the effective modulus of the asperities (and is of the order of one-tenth to one-hundredth of the bulk modulus of the asperity material) and m is a constant ($0< m<1$) which characterises the distribution function of the asperity lengths. Gagi (1978) also stated the importance of using a correct asperity distribution function that can have a considerable effect on the permeability of a fracture.

The permeability of an otherwise impervious cylindrical rock sample of diameter D with a longitudinal crack is given according to Jones (1975) by;

$$k_f = k_{cr}\left(\frac{4e}{\pi D}\right) = \frac{e^3}{3\pi D} \quad (2.17)$$

Where $k_{cr} = e^2/12$ is the intrinsic permeability of the fracture. This relationship is important since most of the crack permeability were tested using cylindrical rock samples with longitudinal crack.

Witherspoon *et al.* (1980) tested the fracture flow for different types of rock with apertures ranging from 4μm – 250 μm and concluded that the cubic relationship stands for all the fractures, having added a factor *f* (1.04 - 1.65) to eliminate the deviations of the parallel plate assumption. Further, it was found that the type of rock does not affect the cubic formula. Tsang (1984) claimed that smaller apertures depress the fracture flow in magnitudes more than 2 when there are contact areas in a

fracture above 30%. Kranz *et al*. (1979) concluded that the overburden pressure and fluid pressure has a great effect on flow through both unjointed and jointed rocks. Walsh (1981) proved mathematically following the experimental investigations available that the cubic root of fracture permeability is linearly related to the effective confining pressure.

2.3.2 *Unsaturated flow modelling*

Since flow through fractures is not always saturated, Indraratna and Ranjith (2001) modelled two-phase stratified flow of air and water through a fracture using the cubic formula. A new apparatus, the High Pressure Two-Phase Tri-Axial apparatus (HPTPTA) was designed to examine the strength and permeability characteristics of fractured and intact rocks under two-phase flow. Figure 2.4 shows the increment of flow rate with inlet fluid pressure and flow rate reduction with confining pressure increments.

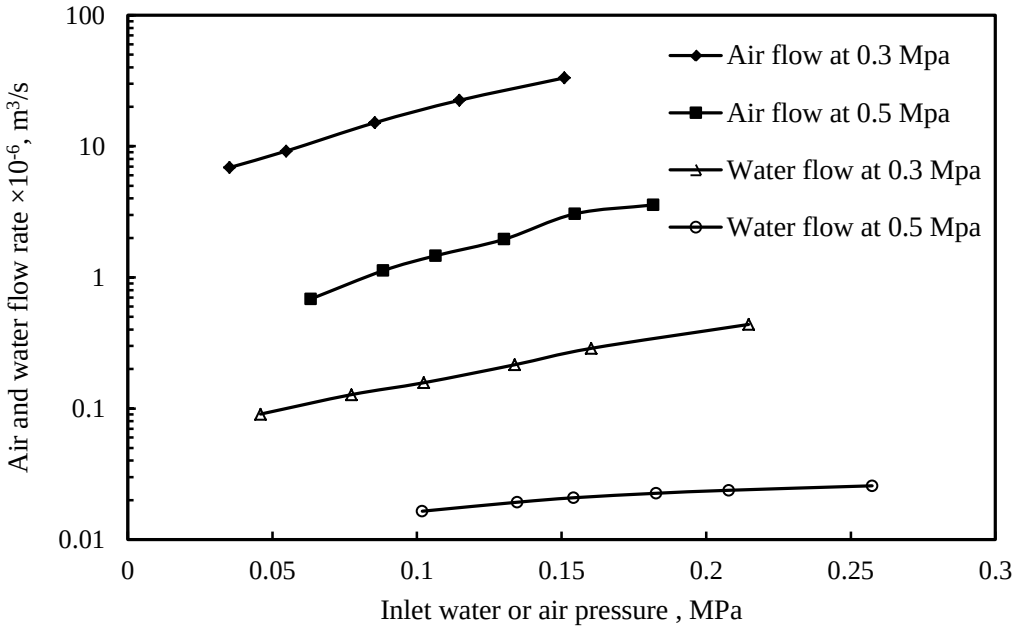


Figure 2.4: Effect of increasing confining pressure and inlet fluid pressure: Single phase air and water flow rates (Indraratna and Ranjith 2001)

With increasing confining pressure, the flow rates of air and water flow were reduced, but at elevated confining pressures the reduction in flow was insignificant (Figure 2.5) because the joint has arrived at its residual aperture state (Indraratna and Ranjith 2001).

The relationship between the Reynolds number and transmissivity of single phase water flow was studied by Ranjith and Darlington (2007), who confirmed that a quadratic relationship exists between them. Moreover, single phase air flow through fractures was tested for the relationship between the Reynolds number and change in pressure, with the results showing that a quadratic relationship applies for confining pressures less than 2.0 MPa and a cubic form of relationship existed for confining pressures greater than 3MPa up to 5MPa. They did not specify what relationship existed between 2MPa and 3 MPa. They also tested the applicability of

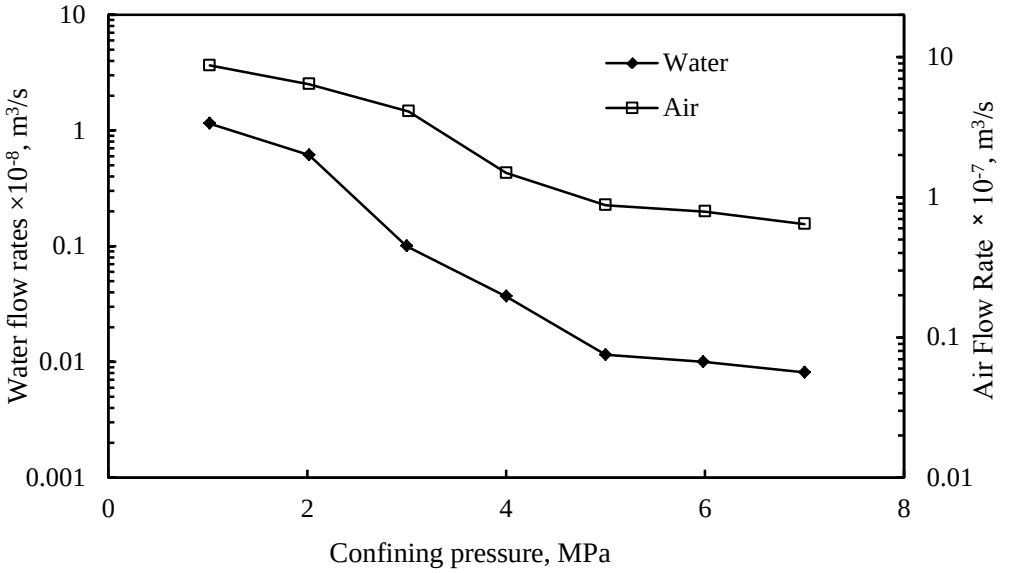


Figure 2.5: Effect of confining pressure on two phase flow rates with a constant inlet fluid pressure of 0.2 MPa (Indraratna and Ranjith 2001)

a Forchheimer-type relationship to single phase air flow and observed that it fits well for low confining pressures. At higher confining pressures the compressibility of air allowed fracture to close and the validity of the Forchheimer-type relationship consistently decreased.

Indraratna *et al.* (2002) developed a simplified stratified flow model for unsaturated flow in jointed rocks. The height of the air phase of the model was said to be a mathematical function of the mechanical deformation of the joint, the compressibility of air and water, and the solubility of air and water. The summation of two phases' heights' or the fracture aperture was related to flow via the cubic formula which confirmed that relationship. Indraratna *et al.* (2003) further studied the stratified two-phase air and water flow and also derived the height of the water-air interface mathematically.

These relationships were formed for flows of a single phase, but in reality, not only a single phase, there may also be more than one fluid flowing at the same time, and therefore a model for multi-phase flow was needed. Indraratna and Ranjith (2001) introduced a new direction for measuring two phase (water and air) flow through a fractured rock specimen, by introducing relative permeability.

$$k_{rw} = -\frac{q_w \mu_w}{k\left(\frac{\partial P}{\partial x} + \rho_w g \frac{\partial z}{\partial x}\right)} \tag{2.18}$$

$$k_{ra} = -\frac{q_a \mu_a}{k\left(\frac{\partial P}{\partial x} + \rho_a g \frac{\partial z}{\partial x}\right)} \tag{2.19}$$

Where q is the flux (m/s), k_{rw} and k_{ra} are the Relative permeability of water and air respectively (Dimensionless), and the 'w' and 'a' suffixes represent water and air respectively. The High Pressure Two Phase Tri-axial Apparatus (HPTPTA) can

measure the individual flow rates of water and air and using that the relative permeabilities could be calculated.

The effect of fluid flow to the mechanical deformation of a fracture was addressed by hydro-mechanical coupling. Indraratna and Ranjith (2001) derived a mathematical model that would consider how the compressibility of fluids, solubility of air and elastic deformability would affect a joint wall. Assuming stratified flow, the equivalent heights of the water and air phases were derived at time *t* relative to a given coordinate system, as follows.

$$h_w(t) = F_I(x,y)_0 - F_B(x,y)_0 - \xi_{wc} \tag{2.20}$$

$$h_a(t) = F_T(x,y)_0 - F_I(x,y)_0 - (\xi_{ac} + \xi_{ad} + \delta_n - \xi_{wc}) \tag{2.21}$$

Where $F_T(x,y)_0$ and $F_B(x,y)_0$ are the initial surface profiles of the top and bottom joint walls respectively, $F_I(x,y)_0$ is interface profile between two fluids, ξ_{wc} is the change in level due to the compressibility of water, ξ_{ac} is the change in level due to the compressibility of air, ξ_{ad} is the change in level due to the solubility of water, and δ_n is elastic deformation of the joint wall.

For insignificant capillary pressure, an expression has been derived for $F_I(x,y)_0$ using the principles of mass and momentum conservation for two phase flow in a given rock joint, as follows:

$$F_I(x,y)_0 = \frac{\tau_{ja}F_B(x,y,\Delta_B) + \tau_{jw}F_T(x,y,\Delta_T) - \tau_{wa}[F_T(x,y,\Delta_T) - F_B(x,y,\Delta_B)]}{\tau_{ja} + \tau_{jw}} \tag{2.22}$$

Where, $F_T(x,y,\Delta_T) = [F_T(x,y)_0 - \xi_{ac} + \xi_{ad} + \delta_n - \xi_{wc}]$

$F_B(x,y,\Delta_B) = F_B(x,y)_0 + \xi_{wc}$

And τ_{ja}, τ_{jw} are the shear stresses acting on joint wall due to air and water respectively, that can be calculated as $\tau = \frac{f\rho v^2}{2}$ where f is friction factor between the joint wall and fluid phase, ρ is density of the fluid phase, v is velocity of the fluid phase, and τ_{wa} is shear stress acting on the water and air interface.

The calculated phase heights were substituted into the following equation in order to assume the phase flux and thereafter the above equation was used so the relative permeability of each phase can be calculated mathematically.

$$q_i = -\frac{{h_i}^2}{12\mu_i}\left\{\frac{\partial P_i}{\partial x} + \rho g \frac{\partial z}{\partial x}\right\} \tag{2.23}$$

The subscript i may represent the water phase or air phase. This relationship has been proven by laboratory experiments. Among the major outcomes of that research Indraratna and Ranjith (2001) found that the single phase flow rates through fractured rock specimens vary linearly against inlet fluid pressures at moderate confining pressures. In the case of two phase flow, an approximately linear relationship (when capillary pressure = 0) was observed between the individual flow components. Furthermore, the relative permeability of the air phase increased exponentially as the P_a/P_w ratio increased, while the relative permeability of water decreased. The opposite trend occurred when the P_a/P_w ratio was increased, and also the two phase flow rates decreased with an increase in the confining pressure.

Indraratna *et al.* (2002) developed a more sophisticated model to determine the height of the interface in stratified two phase flows $F_I(x,y)_0$ given by Equation (2.24).

$$F_I(x,y)_0\Delta_1 - \rho_w g \sin\beta {F_I}^2(x,y)_0 - D = 0 \tag{2.24}$$

Where

$$\Delta_1 = \left[\rho_w g \sin\beta \sum_{k=2}^{3} \Delta_k - \sum_{i=a,w} \tau_{ji} - C\right]$$

Δ_k represents the position on the fracture, i.e. k=2 is bottom and k=3 is the top of the wall

$$C = \left[\frac{1}{l}(v_a - v_w) M \frac{dR}{dx}\right]$$

$$D = e_t[\sigma_1 \sin\beta\cos\beta - \sigma_3 \sin\beta\cos\beta + \tau_{wa}] - \left[\sum_{\substack{k=2 \\ i=a,w}}^{3} \tau_{ji}\Delta_k\right]$$

$$- C \sum_{\substack{k=2 \\ N=1-\eta,\eta}}^{3} N\Delta_k + \rho_w g \sin\beta(\Delta_2\Delta_3)$$

According to the cubic law, the height of the aperture contributes a great deal to the permeability, and since the height of the air phase depends on the mechanical deformation, Indraratna *et al.* (2002) clearly showed that the permeability decreases with an increase of axial pressure, as predicted by the mathematical model. Moreover, the increase of a phase's inlet pressure caused an increase in the flow rate of the same phase, and vice versa.

2.3.3 *Effect of Surface Roughness*

Surface roughness, or the tortuosity of a surface has a significant effect on permeability through fractures. Neuzil and Tracy (1981) considered segments in the fracture to simulate aperture variability and proposed a modified cubic law for flow through rough fractures.

$$Q = \frac{w}{12\mu}\frac{\partial P}{\partial x}\int_0^\infty e^3 f(e)de \tag{2.25}$$

Where, w is the width of the fracture and $f(e)$ is the normalised aperture frequency distribution.

Tsang (1984) showed that the effect of tortuosity increased when there are small apertures in the aperture distribution. The flow rate was said to have reduced by two or three orders of magnitude more than that predicted in the parallel plate model when the contact areas between fracture surfaces rose above 30%. On the other hand, when the aperture variation peaked at larger apertures the effect of tortuosity was smaller.

Brown (1987) also reached the same conclusion from a numerical model that generated realistic rough surfaces using a fractal model of surface topography and put pairs of these surfaces together to form a joint with random aperture distribution. He used Reynolds equation which describes laminar flow between slightly non-planer and non-parallel surfaces.

Ge (1997) used three dimensional aperture distributions to account for the tortuosity of the fracture surface. He assumed single valued functions for both surfaces and the aperture at each point was calculated. His prediction for pressure gradient variation becomes;

$$\frac{\frac{dp}{dx}d_m^3}{12\mu Q_x} = \left[1 + \left(\frac{2\pi d_m \delta}{\lambda}\right)^2 cos^2\frac{2\pi x}{\lambda}\right]^{3/2} \tag{2.26}$$

Where d_m is the mean apparent aperture, δ is the magnitude coefficient of aperture variation, and λ is the wavelength of aperture variation.

2.4 Two-Dimensional Models

When the aperture variability was taken into account for the fracture flow models, the one-dimensional models had limitations in considering the effect of fracture roughness to flow. Therefore researchers developed two dimensional models to simulate real rough rock fracture flows. Two-dimensional models have two types depending on which two-dimensional space is selected.

Type: A

2D modelling from the side view of the fracture

- Zimmerman and Yeo (2013)
- Koyama (2007)
- Indraratna *et al*., (2002)
- Price & Indraratna (2005)

Type: B

2D modelling from top view of the fracture

- Bear *et al.* (1993)
- Kishida et al (2013)

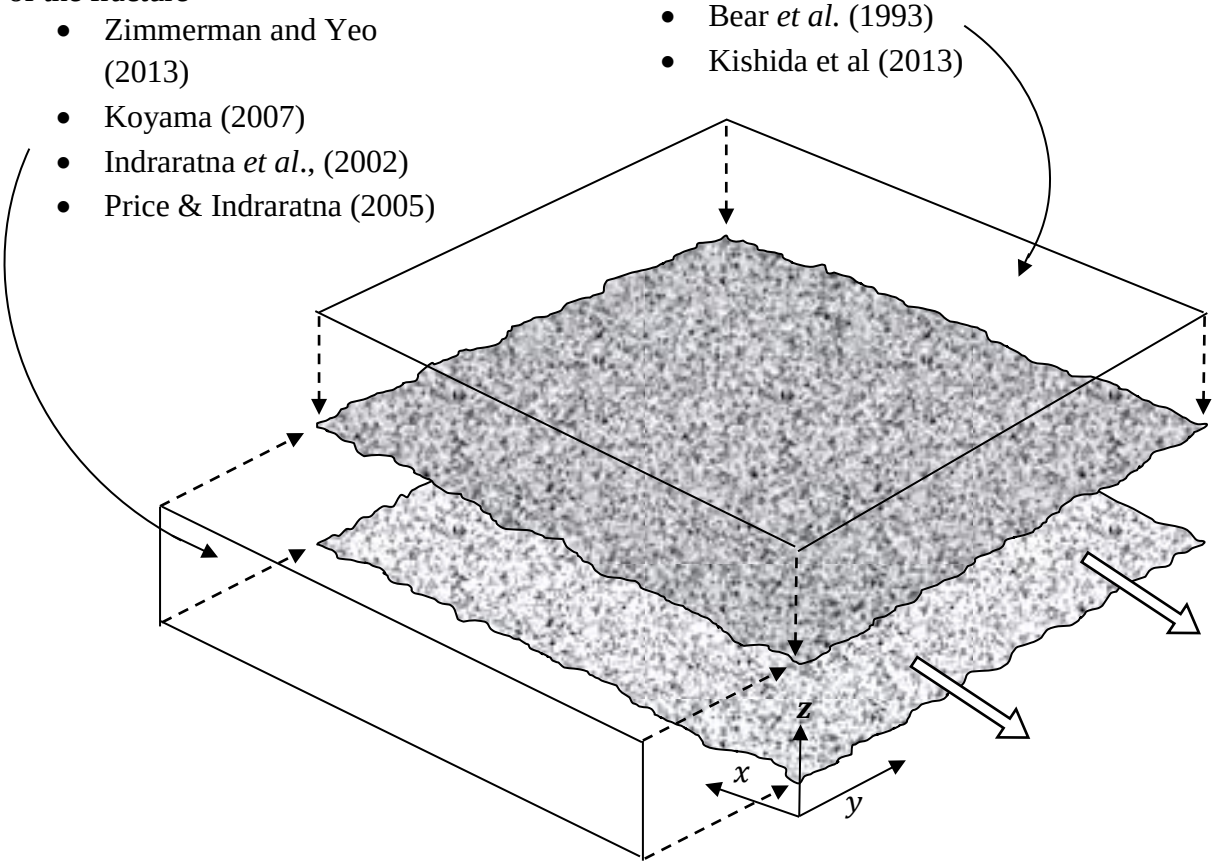


Figure 2.6: Types of 2D flow models according to the selection of 2D space

Figure 2.6 shows a rock fracture and the two views for selecting a 2D space for modelling, as type ‘A’ and type ‘B’. For unidirectional flows, the initial two-dimensional models considered the aperture variability in the direction of flow by taking the height of the aperture as the second dimension (Indraratna et al. 2002; Price and Indraratna 2005; Koyama 2007; Zimmerman and Yeo 2013). Since the height of the aperture is in one dimension, its variability was considered directly in this type ‘A’ models. However, the aperture variations in the *y* direction shown in Figure 2.6 were not considered, and these models also assumed there was no flow component in the *y* direction. Additionally, the type A models cannot handle contacts because a contact will make no flow condition due to the closure of the flow path. To simulate two directional flows, 2D models were developed by using the plan view of the fracture as 2D space for the model (Bear et al. 1993; Kishida et al. 2013). The direction of the aperture was perpendicular to both dimensions in the 2D space of type ‘B’ models, which implies that the aperture would not appear in flow equations that were applicable to this flow problem. For this reason the 3D flow equations were integrated over the aperture direction so that an equivalent 2D equation could be obtained with the aperture as a parameter of the flow.

2.4.1 *2D Models of Type A*

In this type of model the height of the aperture is in the direction of the vertical dimension. This means that variations in the aperture can be directly incorporated into the flow model. These models only account for variations in the aperture in one plane which is parallel to the flow direction, and any variations in the aperture that are lateral to the flow direction were not discussed and the lateral velocity component

perpendicular to the main flow direction was not included in the flow equations, whereas the vertical velocity component was included.

Amadei & Illangasekare (1992) derived a 2D analytical model by assuming one directional flow for steady or transient flows. In their model the aperture was not one separate value, and the user was able to measure the apertures at preferred

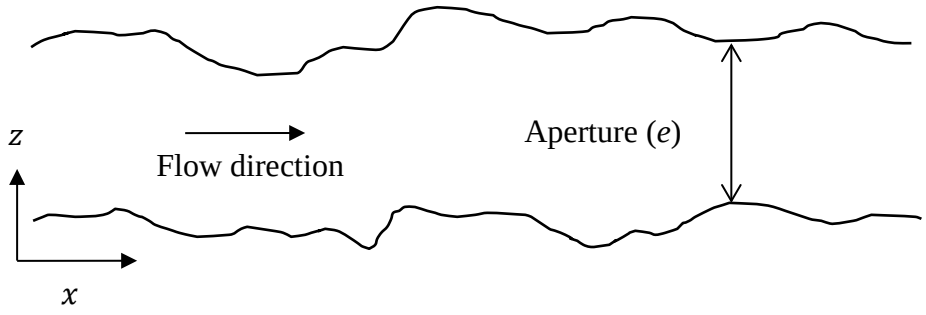


Figure 2.7: Type A models' 2D space

intervals and then calculate the associated flow behaviour. Indraratna *et al*. (2002) and Price and Indraratna (2005) suggested using Fourier analysis to describe the irregularity of the fracture aperture. Fourier coefficients were calculated using the scanned joint surfaces and the variations of the fracture aperture were obtained as Fourier functions. Zimmerman and Yeo (2013) used a 2D rock fracture viewed from one side of the fracture and proved that the Navier-Stokes equation can be linearised into the Stokes equation by neglecting the advection acceleration terms compared to the viscous terms, provided the Reynolds number of the flow is less than about 10. Koyama (2007) modelled flow and particle transport in rock fractures during shear by looking at the fracture from one side and then using the Reynolds equation. Here the Navier-Stokes equations were solved using commercial software and the simulations were compared with laboratory shear-flow-tracer tests to demonstrate that the Reynolds equation overpredicts the flow rate by roughly 5-10% compared to

the Navier-Stokes analysis. Zimmerman *et al.* (2004) conducted laboratory measurements and Navier-Stokes simulations on fracture flow to study the non-linear regimes of fracture flow. They discovered a weak inertia regime within the Reynolds number range of 1-10, although the influence was significantly less for the Forchheimer type regime beyond a Reynolds number of 20.

2.4.2 *2D Models of Type B*

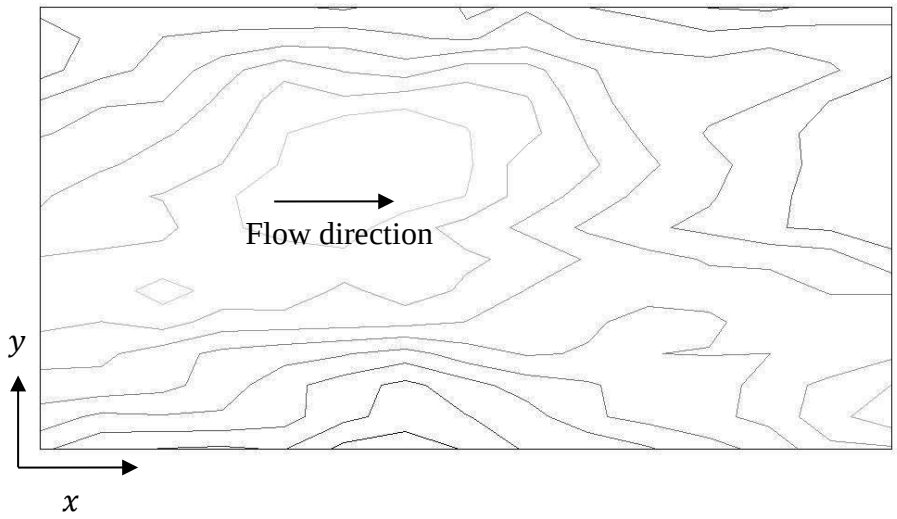


Figure 2.8: Type B models' 2D space. Apertures shown as contours

The latter approach can be used to define the spatial distribution of flow in a 2D fracture where velocity perpendicular to the plane of the aperture was assumed to be negligible compared to the flow that occurs in the lateral and longitudinal directions of the fracture. Unlike type A 2D models, Bear *et al.* (1993) modelled the flow in a non-deforming fracture through the plan view (Figure 2.8) by integrating the three-dimensional Navier-Stokes equation in the direction of the aperture height. Kishida *et al*. (2013) used the same method to develop a 2D model for non-deforming rock

fracture walls that was solved numerically using the highly simplified marker and cell (HSMAC) method. Both models assumed that the fracture aperture was constant for the integration. Since the limits of the integration are the profiles of the fracture walls, if the fracture is assumed to be deformable, the integration limits also become functions of time. Therefore, in order to use the derived 2D model for deformable fractures, it should be considered when the 3D flow equations are to be integrated.

2.5 Numerical Simulations of Fracture Flow

Researchers used fluid flow solvers to simulate flow through rock fractures. Pruess and Tsang (1990) used the general purpose simulator MULKOM to simulate two phase flow through rock fractures. Khalili-Naghadeh and Valliappan (1991) developed finite element implicit coupled double porosity model for fissured porous media. Koyama (2007), and Koyama *et al.* (2008) used the COMSOL finite element solver to simulate fracture flows while solving the Reynolds equation. These commercial solvers can simulate the contacts in the apertures so a flow domain should be constructed in the solver design environment. These solvers are designed to treat the local boundaries, and contacts act as local boundaries in the flow domain, however, to study the contact formation and evolution with normal stress increments, the flow domain boundaries should be treated as moving boundaries. This task should be achieved by writing user defined functions for the boundary walls, which is why most 3D simulations were only carried out for constant apertures. Using 3D simulations to calculate the permeability of rock fractures would be a difficult task, because, after the apertures have been scanned, each fracture domain should be constructed for the simulation. Two dimensional type B simulations have an

advantage over 3D simulations in that constructing a domain is not necessary while the simulation is still considering the 3D features of the fracture.

2.6 Effects of Contacts to the Fracture Flow

When a fracture contains apertures in a range of magnitudes, the smaller apertures tend to become contacts for flowing fluids. Two surfaces in contact cannot be declared physically closed without any other reference, because, there is always a distance between two molecules of any matter. The two surfaces can be closed for a particular fluid according to fluid properties such as particle dimensions and viscosity. Furthermore, fluids flow selectively when there are optional paths available, so if adjacent apertures are larger, the smaller apertures may become closed to the fluid. Aperture contacts act like obstacles in general stream flows and they also affect the flow pattern and flow rates too. General one dimensional models cannot handle apertures as well as type A two dimensional models, because in both cases the presence of zero apertures causes a no-flow condition.

When there are asperities present in a fracture, the permeability of the fracture may differ from the permeability of a fracture with no asperities (k). Iwai (1976) derived ⟨k⟩ the flow conductance as;

$$\langle k \rangle = \frac{1-\alpha}{1+\alpha} k \tag{2.27}$$

Where α = ratio of the contact area to the total area of the fracture. Walsh (1981) derived the same Equation (2.27) using the Maxwell effective medium approximation, and proved the relationship is accurate for contact areas up to at least 25% by considering circular shape contacts.

Zimmerman *et al.* (1992) used boundary element calculations, analogue electrical conductivity measurements, and effective medium approximation to study the flow of rock fractures having contacts of different shapes, sizes, and area percentages. They extended the Maxwell-Walsh approach to randomly oriented elliptical shaped contacts and modified it to:

$$\langle k \rangle = \frac{1 - \alpha\beta}{1 + \alpha\beta} k \tag{2.28}$$

Where $\beta = (1 + \gamma)^2/4\gamma$, γ is the ratio of the minor to major axis of the ellipse which becomes 1 for a circle. They verified this model for γ = 0.2 and $\alpha \leq 0.05$. Furthermore, while using the boundary element method it was discovered that for irregular shaped contacts the permeabilities were lower than up to 30% of the Walsh predictions. Assuming an equivalent ratio γ, Equation (2.28) produced more acceptable permeabilities than the Walsh predictions.

Li *et al.* (2008) studied the transmissivity of artificial rock fractures with circular shaped contacts in a number of patterns by assuming cubic law. They suggested an enlargement factor for the Equation (2.27) to multiply the contact area ratio (α) in the range 1 – 2 to predict permeability / transmissivity of rock fractures. Koyama *et al.* (2009) solved Reynold's equation using a commercial FEM solver COMSOL multi-physics to simulate rock fracture flow. Flow domain was generated with contacts as local boundaries and finite element mesh was generated with much finer meshes around the contact areas. These simulations were repeated by replacing contacts with smaller apertures (1μm) and it was then found that the flow rates were almost equal, but the flow patterns with contacts provided physically meaningful results.

2.7 Experimental Rock Fracture Flow Testing

Laboratory rock fracture flow tests were conducted using triaxial apparatus or direct shear testing apparatus. For any type of research, laboratory experiments are preferred when behaviours need to monitor in a controlled environment. In reality many parameters can affect a natural behaviour and it is not easy to distinguish the influence of some of the critical parameters. Laboratory experiments enable the exact influence of selected parameters to be investigated in a preferred way, which is why laboratory work plays a major role in any research project.

Laboratory test results are used to verify a suggested phenomenon or investigate the behaviour of control parameters in order to suggest a new model. The latter is referred to as empirical modelling and is widely used for very complex behaviours which are practically unrealistic to model using constitutive relationships. Although the cubic formula was derived by assuming a parallel plate to model rock fracture flow, the roughness of the fracture creates uncertainties about the cubic formula predictions. Usually the cubic formula over predicts real behaviour and laboratory investigations were carried out to find the limitations of the cubic formula (Witherspoon et al. 1980; Bart and Shao 1998; Bart et al. 2004).

The behaviour of rock fracture flow was investigated with normal loading because flow is more sensitive to the aperture than other parameters and normal loading causes aperture deformation. Rectangular specimens (Iwai 1976; Lee and Cho 2002) and cylindrical specimens (Kranz et al. 1979; Indraratna et al. 1998; Indraratna et al. 2002; Indraratna et al. 2003) were used in hydro-mechanical coupled flow testings for rock fractures.

2.7.1 *Flow tests on horizontal rock fractures*

Cylindrical and rectangular specimens can both be tested horizontally (Figure 2.9) and the tests were mainly carried out on direct shear apparatus. Iwai (1976) tested rectangular and cylindrical specimens of different rock types to check the validity of the cubic law. The cylindrical specimens were fractured perpendicular to the axis and radial flow was tested by injecting water from the centre. The test results showed that the cubic law was valid for the specimens tested in both straight and radial flows. Tsang and Witherspoon (1981) tested cylindrical specimens with horizontal fracture for radial flow and concluded that a simple smooth parallel plate representation would probably be inadequate to analyse fluid flow through a deforming rock fracture. Raven and Gale (1985) used a cylindrical specimen with a sub-axial fracture to test unidirectional flow behaviour under cyclic loading and also observed how the scale of the specimen affected flow. Schrauf and Evans (1986) used rectangular specimens to test unidirectional and radial gas flow through rock fractures and found that the flow rate of gas was inversely proportional to the fourth power of the aperture. Lee and Cho (2002) also used rectangular rock specimens to investigate the coupled hydro-mechanical rock fracture flow models.

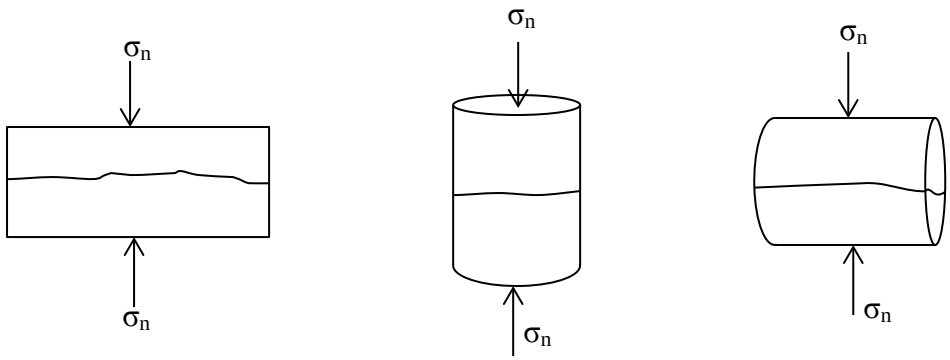


Figure 2.9: Arrangements of specimens in direct shear apparatus. (a) rectangular specimens. (b) cylindrical specimens – radial fracture. (c) cylindrical specimens – axial fracture.

2.7.2 *Flow tests on vertical rock fractures*

Cylindrical specimens can be tested in the triaxial apparatus for vertical single fracture flows (Figure 2.10). A sub-axial fracture is usually tested within a triaxial stress environment because it simulates the stress environment underground much better.

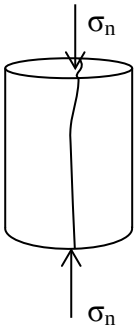


Figure 2.10: Rock fracture arrangement in triaxial apparatus for flow tests

Indraratna and Ranjith (2001) measured two phase flow parameters using high pressure triaxial testing for cylindrical rock specimens. The relative permeability of a

phase was found to be proportional to the ratio of phase inlet pressure to the other phase's inlet pressure. Two phase flow rates decreased with confining pressure and reached an asymptotic state at higher confining pressures. Ranjith and Darlington (2007) experimented with single phase flows of water and air through a vertical fracture induced onto cylindrical rock specimens to study the nonlinear behaviour of fracture flow using triaxial apparatus. Zhang and Nemcik (2013) tested cylindrical specimens with a sub axial fracture to study the friction factor introduced to the cubic law in order to eliminate errors due to fracture roughness. They found that the relative roughness of the fracture affected the friction factor and when the Reynolds number was greater than unity, the difference in the friction factor induced by the relative roughness was reduced.

2.7.3 *Aperture Measurement Techniques*

Rock fracture apertures can be measured to obtain the mechanical aperture of a fracture using direct methods such as feeler gauges, fluorescent dyes, and the impression packer method. Hydraulic apertures were calculated indirectly using the volumetric flow rates back calculated by assuming the cubic formula. The cubic predictions were over estimations, so more descriptive techniques were used to measure the aperture distribution in the laboratory. Further to the cubic back calculation, tracer tests (equivalent mass balance aperture method, equivalent frictional loss aperture) are other indirect methods that can be used to measure fracture aperture. Zimmerman *et al.* (2004) used a profilometer to acquire the surface profiles of both opposing fracture surfaces, and fracture apertures were obtained using the surface profile data in a mesh generator. The two surfaces were brought

closer the same as preparing the fracture from two walls, and the space between the two surfaces was used as the fracture aperture.

Replicating the fracture aperture is another technique which directly measures the fracture aperture. The gaps between two fracture surfaces are filled with an epoxy resin solution which solidifies fast, and then the replicated aperture can be used to measure the aperture distribution. Price and Indraratna (2005) used a laser scanner to scan the replicated aperture and produced the aperture distribution by taking the difference between the replica surface and the fracture surface from which the replica was created.

2.8 Rock Fracture Deformation

Studying the normal deformation of rock fractures is important in mining because the fracture normal stress applied to rock fractures increases with the depth of the mine; this means that the permeability of rock fractures is also a function of the depth. Predictions for the normal deformation of rock fractures were made mathematically and experimentally.

Several researchers (Bandis *et al.*(1983); Indraratna and Oliveira (2010); Oliveira and Indraratna (2010)) studied the deformation of clean or in-filled joints under stress. The contribution to the shear and normal deformation by the infill material is important because the shear strength and joint stiffness change with the thickness of the infill. In accordance with the stiffness of the joint, the deformations that take place and the applied stresses contribute directly to the deformations, but with fluid inside a fracture, the pore pressure acts against the surrounding total stress, and the resulting effective stress is responsible for the fracture deformation (Nur and Byerlee 1971; Garg and Nur 1973; Carroll 1979; Dhowian 1980; Khalili

and Valliappan 1996). Combined theories of deformation and flow through rock joints, coupled hydro-mechanical (poro-elastic) models, have been introduced (Nguyen and Selvadurai 1998; Bart et al. 2004; Zhou et al. 2008; Tao and Ghassemi 2009).

When there is a pore fluid inside a joint, the stress applied to the joint will be affected by the pore pressure. The actual stress that applies to the joint will be given by the effective stress;

$$\sigma^{/} = \sigma - \varphi u_w \tag{2.29}$$

Here, $\sigma^{/}$ is the effective stress, σ is applied stress, φ is a constant, and u_w is the pore pressure. Experimental strength measurements have shown that φ is approximately close to unity, according to Nur and Byerlee (1971).

2.8.1 *Deformation and Effective Stress*

By considering a fracture inclined θ to the horizontal where the walls are perpendicular to the X-Z plane (Figure 2.11), and assuming an area of 'a' on the joint wall and σ_z is greater than σ_x, the forces acting on the joint wall were given by Farmer (1983) as;

$$\sigma_n a = (\sigma_z cos\theta - \tau_{zx} sin\theta) acos\theta + (\sigma_x sin\theta - \tau_{xz} cos\theta) asin\theta \tag{2.30}$$

$$\tau a = (\sigma_z sin\theta - \tau_{zx} cos\theta) acos\theta - (\sigma_x cos\theta - \tau_{xz} sin\theta) asin\theta \tag{2.31}$$

Since $\tau_{zx} = \tau_{xz}$, Equations (2.30) and (2.31) can be reduced to Equations (2.32) and (2.42) as follows.

$$\sigma_n = \sigma_z cos^2\theta + \sigma_x sin^2\theta - 2\tau_{xz} sin\theta cos\theta \quad (2.32)$$

$$\tau = (\sigma_z - \sigma_x) sin\theta cos\theta - \tau_{xz}(sin^2\theta - cos^2\theta) \quad (2.33)$$

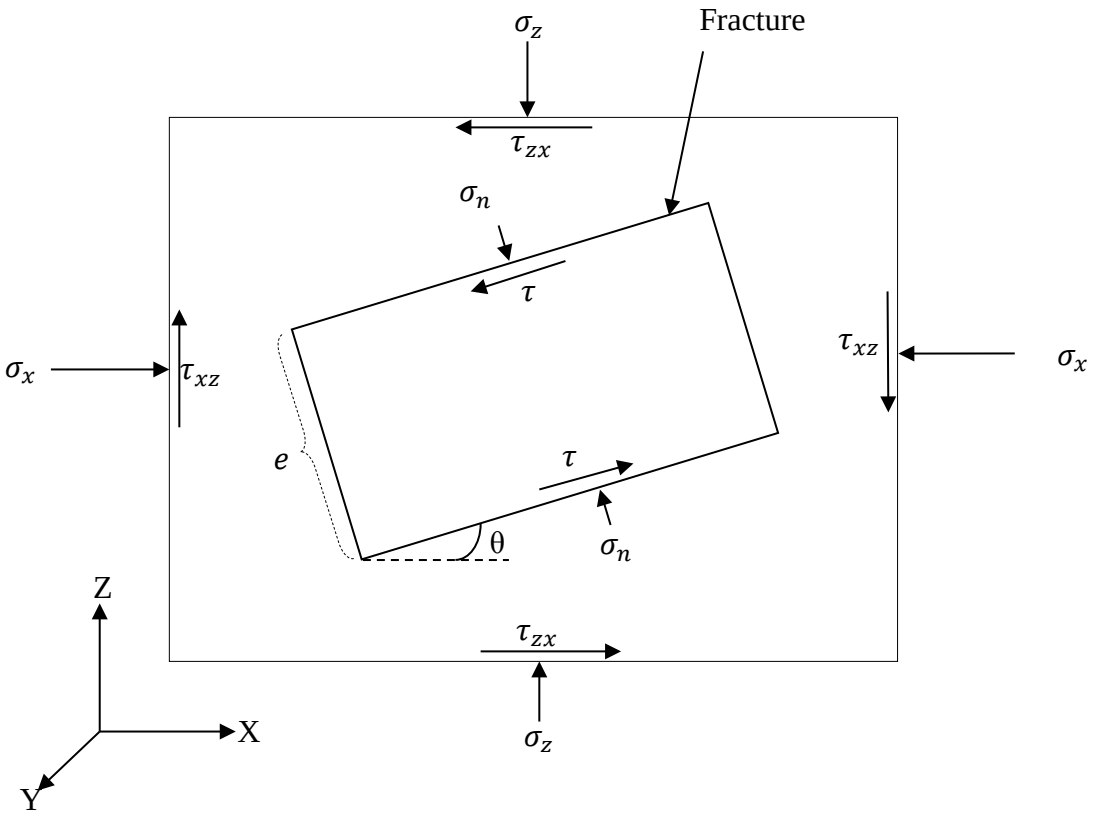


Figure 2.11: Stress environment of a fracture

When the X axis and Z axis are assumed to be the principal stress axes, the joint normal stress and joint shear stress are given by Equations (2.34) and (2.35).

$$\sigma_n = \sigma_1 cos^2\theta + \sigma_3 sin^2\theta \quad (2.34)$$

$$\tau = (\sigma_1 - \sigma_3) sin\theta cos\theta \quad (2.35)$$

σ_n is the normal stress acting on the fracture, and τ is the shear stress acting on the fracture.

2.8.2 *Joint Stiffness*

Rock joints tend to close/open or shear under applied normal or shear stresses, respectively. Generally the stiffness is the change in applied stress per unit change in deformation (Yoshinaka and Yamabe 1986; Xiao et al. 1994).

$$k_n = \frac{d\sigma_n}{d(\Delta v)} \tag{2.36}$$

$$k_s = \frac{d\sigma_s}{d(\Delta d_h)} \tag{2.37}$$

Here k_n is the joint normal stiffness and k_s is the joint shear stiffness, σ is the applied stress, Δv is the vertical deformation, and Δd_h is the horizontal deformation.

2.8.2.1 Joint Normal Stiffness

A single value cannot be defined for the joint normal stiffness k_n since joints are deformed non-linearly with applied normal stress. Bandis *et al.* (1983) proposed a hyperbolic model (Equation (2.38)) to describe the normal load and displacement behaviour of rock joints according to the hyperbolic stress – strain curves of rocks under triaxial compression, as proposed by Kulhawy (1975).

$$\sigma_n = \frac{\Delta v}{a - b\Delta v} \tag{2.38}$$

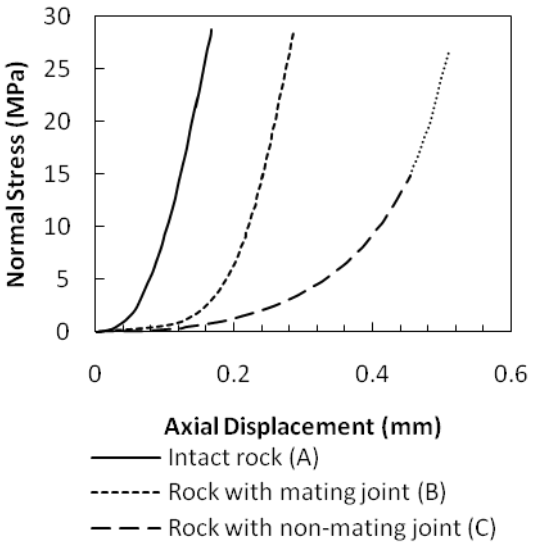
Where a and b are empirical constants. Gens *et al.* (1990) used similar model for normal stress – normal displacement relationship to develop elastoplastic constitutive law for rock joints.

When normal stress tends to infinity, Bandis *et al*. (1983) showed that the limiting value for Δv will be a/b, so a/b will be the maximum closure v_m of the joint where the stiffness becomes infinity. From the derivative of Equation (2.38) it can be seen that the initial joint normal stiffness k_i is $1/a$. Then k_n at any level of σ_n (Equation (2.39)) will be given by the derivative of the Equation (2.38) substituted $a = 1/k_i$ and $b = 1/k_i v_m$. Δv should be obtained from Equation (2.38).

$$k_n = \frac{k_i}{\left(1 - \frac{\Delta v}{v_m}\right)^2} \tag{2.39}$$

Goodman (1974; 1976) proposed that maximum joint closure (v_m) should be less than the aperture of a joint, and the closure of the fracture under normal stress was obtained by subtracting the normal deformation of the intact rock (Figure 2.12). Tsang and Witherspoon (1981) concluded from the experiment results that a fracture cannot be completely closed unless the applied normal stress is extremely high, which agrees with the hyperbolic model explained above.

(a)



(b)

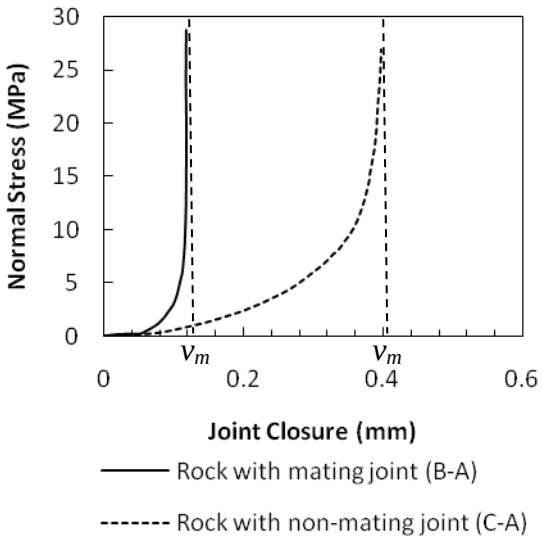


Figure 2.12: (a) Normal stress vs. axial displacement intact and jointed rock, (b) Normal stress vs joint closure of fractured rock (Goodman 1976)

Brown and Scholz (1986) derived a semi logarithmic relationship for joint deformation and normal stress by following the force-deformation relationships of the elastic contacts of spherical bodies.

$$\Delta v = A + B \ln(\sigma_n) \tag{2.40}$$

A and B are material constants that depend on the geometry of the surface profile. Bandis *et al.* (1983) also suggested a semi log stress-deformation model for dislocated joints similar to that shown in Equation (2.41) for dislocated joints.

$$\ln(\sigma_n) = p + q\Delta v \tag{2.41}$$

The initial normal stress given by the constant p, and q is a function of joint normal stiffness and deformation.

2.9 Numerical Methods for Solving Rock Fracture Flow Models

When rock fracture tortuosity and multi-dimensional flows are considered, numerical solutions were needed to solve the non-linear equations and simulate the flow behaviour in a macroscopic view. Since the Navier-Stokes equation (Equation (2.1)) is difficult to solve even when using numerical techniques, researchers have often simplified these equations.

2.9.1 *Solutions for Pressure-Velocity Coupled Rock Fracture Flow Models*

Two-dimensional flow models are non-linear partial derivative equations (PDE) that cannot be solved explicitly. The usual approach is to use a numerical method and solve them in a descretised form to find an approximate solution. For fracture flows, the fluid pressure cannot be assumed to be homogeneously distributed because the

fracture apertures are uneven, and therefore the velocity and pressure should be calculated to solve the flow models. The Navier-Stokes equations consist of momentum equations and a continuity equation, and since there is no separate equation to calculate fluid pressure, additional numerical methods are needed to couple the pressure and velocity. Patanker and Spalding (1972) introduced SIMPLE (Semi Implicit Method for Pressure Linked Equations) algorithm to solve such flow equations iteratively. SIMPLE algorithm was later modified for better and faster solutions by Patankar (1980) to SIMPLER (SIMPLE Revised), and by Van Doormaal and Raithby (1984) to SIMPLEC (SIMPLE Consistent). Issa (1986) developed a pressure-velocity coupled algorithm to solve flow equations called PISO (Pressure Implicit with Splitting of Operators), that has two steps called a predictor and a corrector to solve unsteady flow equations non-iteratively. This method has been adapted to solve steady flow equations iteratively.

2.9.2 *Partial Derivative Equations and their Discretisation*

In order to find a numerical solution for PDEs, the equations should be descretised. Finite Volume Method (FVM) is a popular method for discretising flow equations because it involves integrating the governing equation over a control volume of the domain, and then the integrated equations are approximated using a suitable differencing scheme. Differencing schemes are methods that approximate partial derivatives in a PDE. The first derivative of a function *f* is defined as;

$$\frac{\partial f}{\partial x} = \lim_{h \to 0} \left(\frac{f(x+h) - f(x)}{h} \right) \tag{2.42}$$

These differencing schemes use a small value for h rather than going to its limit of zero and therefore the approximation is not the exact derivative.

$$\frac{\partial \mathrm{f}}{\partial x} \approx \frac{f(x+h)-f(x)}{h} \tag{2.43}$$

The three basic differencing methods are;

1. Forward differencing method

$$\frac{\partial \mathrm{f}}{\partial x} = \frac{f(x+h)-f(x)}{h} \tag{2.44}$$

2. Backward differencing method

$$\frac{\partial \mathrm{f}}{\partial x} = \frac{f(x)-f(x-h)}{h} \tag{2.45}$$

3. Central differencing method

$$\frac{\partial \mathrm{f}}{\partial x} = \frac{f(x+h/2)-f(x-h/2)}{h} \tag{2.46}$$

The difference between the derivative and the approximation is the error in the differencing method which can be found using the Taylor expansion for $f(x+h)$.

$$\frac{\partial \mathrm{f}}{\partial x} - \frac{f(x+h)-f(x)}{h} = error \tag{2.47}$$

Forward and backward differencing methods give a first order error (i.e. error is proportional to the magnitude of h) and the central differencing scheme gives second order error (i.e. error is proportional to the square of the magnitude of h). This means the central differencing scheme is accurate for smaller values of h.

In the finite volume method discretisation, the forward and backward differencing methods are used selectively together in a scheme called upwind where the neighbouring value is selected from upstream of the flow. That value can be

f(x+h) or *f(x-h)* according to the direction of flow. Although the upwind scheme has a first order error, it is more stable than the central differencing scheme. The QUICK differencing scheme uses a third order interpolation using three upstream neighbours for the upwind method, which enhances the accuracy of the upwinding method. However, using more neighbours for the approximations in higher order schemes causes problems when implementing boundary values.

2.9.2.1 SIMPLE Algorithm

The SIMPLE algorithm should be initialised with a guessed pressure field where the velocities are calculated by solving the descretised momentum equations and then using them to correct the guessed pressure, and then the velocities calculated initially using the momentum equations will also be corrected. These steps are repeated until the corrections for velocity and pressure are negligible, and then the iteration process is stopped. Details of this method are explained in Chapter 3.

2.10 Summary

The chapter outlined the rock fracture flow modelling techniques used to predict rock fracture flows. Initial one dimensional flow models beginning with the cubic formula, and later modifications and developments that have taken place were explained briefly. The need to model two dimensional flow models and two types of two dimensional models developed to simulate fracture flows were discussed. The normal deformation of rock fractures under applied normal stress that were modelled and experimented by previous researches, including a brief introduction of fracture deformation and modelling, was presented in this chapter. Numerical simulations

carried out for fracture flows and experimental investigations for rock fracture flows were also discussed. Finally, the numerical methods that can be used to solve pressure velocity coupled flow models in rock fracture flow studies were introduced. They will be used in Chapter 4 to solve the two-dimensional flow model proposed in Chapter 3.

Chapter 3 A Two-Dimensional Flow Model for Rough Deformable Rock Fractures

3.1 Introduction

The origin of rock fractures, rock fracture classifications, and an introduction to rock fracture flow was presented in Chapter 2. The surfaces of real rock fractures are usually rough in nature. The roughness may vary from smooth to rough, and is quantified by the joint roughness coefficient (JRC). The aperture is the space created between the two surfaces of a rock fracture and it too becomes irregular due to the roughness of the fractured surfaces. When a fluid is transported through this kind of path, the flow pattern is governed by the flow path geometry, but since apertures vary spatially, the use of only one equivalent aperture to model flow through a fracture leads to errors in prediction. Development of accurate 3D models can be so cumbersome because of the need to capture highly variable and anisotropic fracture details through extensive geological mapping and site investigations, and when these features are properly captured the programme execution time invariably becomes unacceptably long. Therefore, researchers are then forced to make simplifications in order to attain reasonably fast convergence, and in doing so, these 3D models often do not yield results that offer significantly better accuracy than the easy-to-use plane

strain or 2D models (Indraratna and Ranjith, 2001; Zimmerman and Yeo, 2013). Rock fractures are active due to the changing stress environment and undergo deformations upon loading that directly affect the characteristics of the fracture flow as the flow path changes. This chapter presents a two dimensional flow model developed for a normally deformable rock fracture, followed by a numerical solution.

3.2 Development of the Mathematical Model

Two dimensional (2D) mathematical models were developed to account for the tortuous flow paths found in rock fracture flow studies. 2D models were developed by considering a rock fracture by its longitudinal cross section or its plan view. Since the longitudinal cross section cannot support aperture tortuosity in the lateral direction of flow, the plan view was used to model the flow behaviour in 2D.

Figure 3-1 shows a typical rough rock fracture that conducts a fluid flow. Since $[\acute{X}, \acute{Y}, \acute{Z}]$ is the global coordinate system, a local coordinate system [x,y,z] was defined with the [xy] plane parallel to the plane of the fracture surface.

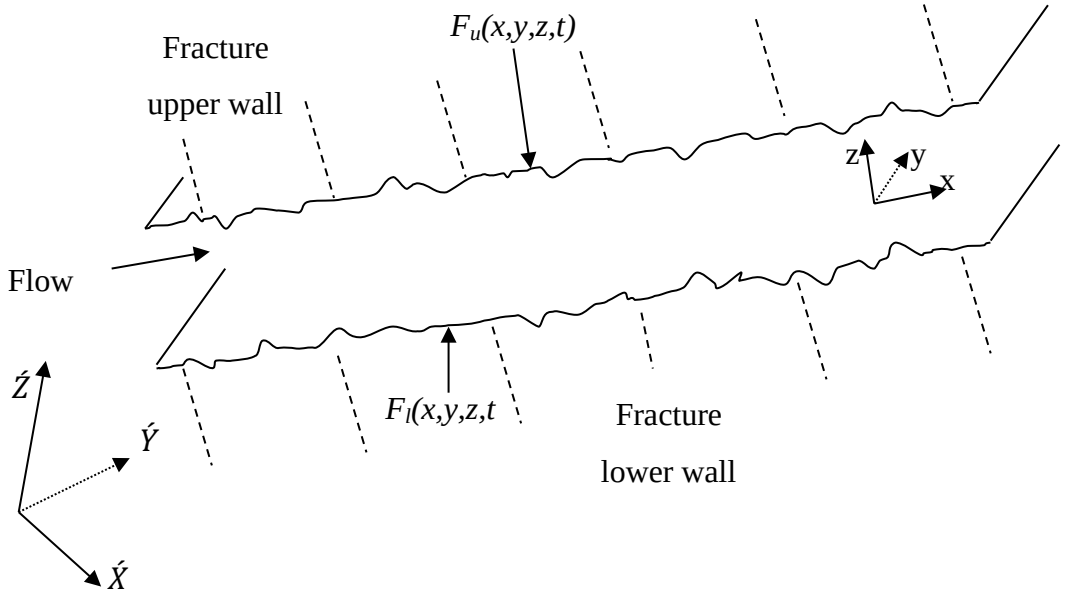


Figure 3-1: Flow through a rough rock fracture

3.2.1 *Conservation of Momentum*

The three dimensional linear momentum conservation of an incompressible fluid flow is given by the Navier – Stokes Equation (2.1) as follows:

$$\frac{\partial(\rho \boldsymbol{V})}{\partial t} + \boldsymbol{\nabla}.(\rho \boldsymbol{V}\boldsymbol{V}) + \boldsymbol{\nabla} p - \rho g \boldsymbol{\nabla} Z - \boldsymbol{\nabla} \cdot \bar{\bar{\mathbf{T}}} = 0 \tag{3.1}$$

Where $\boldsymbol{V}$ is the velocity vector, ρ is the fluid density, p is the pressure, g is the gravitational acceleration, and $\bar{\bar{\mathbf{T}}}$ is the stress tensor given by:

$$\bar{\bar{\mathbf{T}}} = \mu\left[\boldsymbol{\nabla}\mathbf{V} + (\boldsymbol{\nabla}\mathbf{V})^{\mathrm{T}} - \frac{2}{3}(\boldsymbol{\nabla} \cdot \mathbf{V})\bar{\bar{\mathbf{I}}}\right] \tag{3.2}$$

Here $\bar{\bar{\mathbf{I}}}$ is the unit tensor and μ is the dynamic viscosity of the fluid. Assuming the fluid as a homogeneous, Newtonian, incompressible liquid, Equation (2.1) can be expressed by:

$$\frac{\partial(\rho \boldsymbol{V})}{\partial t} + \boldsymbol{\nabla}.(\rho \boldsymbol{V}\boldsymbol{V}) + \boldsymbol{\nabla} p - \rho g \boldsymbol{\nabla} Z - \mu \nabla^2 \boldsymbol{V} = 0 \tag{3.3}$$

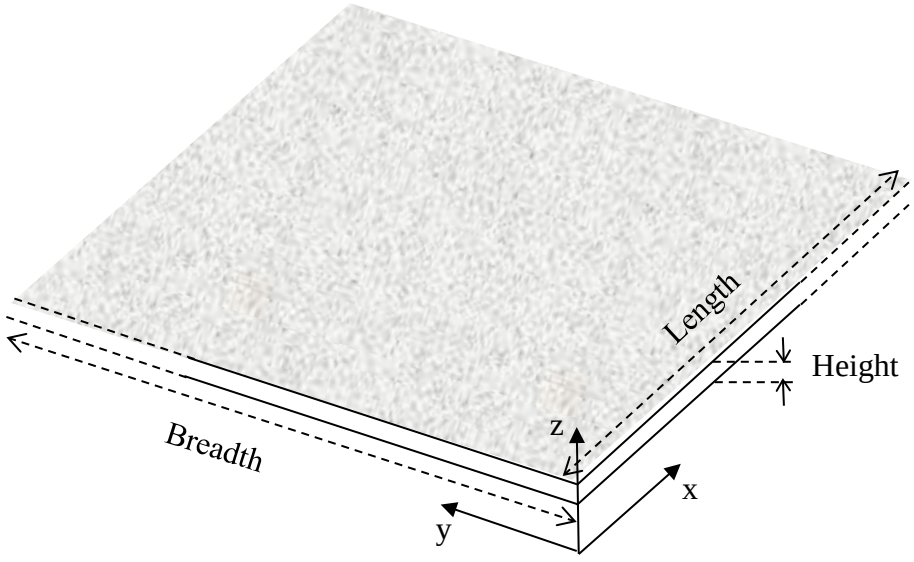


Figure 3-2: Dimensions of a rock fracture

The distribution of velocity and pressure in three-dimensional space is given by Equation (3.3). The flow domain of a rock fracture can be considered as a thin duct that is longer and wider than the height (Figure 3-2). Since rock fracture flows are slow, turbulent flows are not expected and therefore, the velocity in the z direction is negligible most of the time.

The height of the rock fracture or the aperture was not a constant over the two dimensional space when compared to a thin duct, so the z dimension of the equation governing the flow cannot be omitted when considering two dimensional flow. To account for variations in the aperture over the space and to develop an equivalent two dimensional flow model, Equation (3.3) can be integrated in the z direction by using upper and lower walls as limits. Bear *et al.* (1993) and Kishida *et al.* (2013) modelled water flow through non-deformable rock fractures using this method and by considering stationery fracture walls. Since rock fractures deform over time according to normal loading, the fracture walls cannot be assumed to be stationery, so in this study, integration in the z direction was done by assuming the fracture walls deformed normally. If F_l and F_t are the surface profiles of lower and upper fracture walls respectively, then Equation (3.3) can be integrated as;

$$\int_{F_l}^{F_t} \left(\frac{\partial(\rho \boldsymbol{V})}{\partial t} + \boldsymbol{\nabla}.(\rho \boldsymbol{V}\boldsymbol{V}) + \boldsymbol{\nabla} p - \rho g \boldsymbol{\nabla} Z - \mu \nabla^2 \boldsymbol{V} \right) dz = 0 \tag{3.4}$$

When integrating a differential equation, if the limits of the integration are functions of the variables of the differential equation, Leibnitz integral rule should be adopted as follows:

$$\int_g^h \frac{\partial}{\partial x_i} F(x_1,..x_i..x_n)dx_j$$

$$= \frac{\partial}{\partial x_i}\int_g^h F(x_1,..x_i..x_n)dx_j + F(g)\frac{\partial}{\partial x_i}g - F(h)\frac{\partial}{\partial x_i}h \tag{3.5}$$

Here, g and h are functions of x_i and some other variables. Each component of the equation (3.4) was integrated individually by using the Leibnitz integral rule. Integration of the first term in equation (3.4) is;

$$\int_{F_l}^{F_t} \frac{\partial(\rho \boldsymbol{V})}{\partial t}dz = \frac{\partial}{\partial t}\int_{F_l}^{F_t}(\rho \boldsymbol{V})dz + \frac{\partial F_l}{\partial t}\cdot \rho \boldsymbol{V}|_{f_l} - \frac{\partial F_t}{\partial t}\cdot \rho \boldsymbol{V}|_{f_t} \tag{3.6}$$

In the above, $F_l = z_l + f_l$ and $F_t = z_t + f_t$ where z_l and z_t are the z coordinates of the profiles of the bottom and top wall at a particular point in (x,y) space given by f_l or f_t , respectively. Assuming the no slip boundary conditions at the walls for the fluid, equation (3.6) will then become;

$$\int_{F_l}^{F_t} \frac{\partial(\rho \boldsymbol{V})}{\partial t}dz = \frac{\partial}{\partial t}\int_{F_l}^{F_t}(\rho \boldsymbol{V})dz \tag{3.7}$$

The density and pressure of the fluid was assumed to be constant in the z direction because the fracture aperture was much smaller than the two other fracture dimensions. By substituting the depth average velocity:

$$\int_{F_l}^{F_t} \frac{\partial(\rho \boldsymbol{V})}{\partial t}dz = \frac{\partial}{\partial t}(\rho e\overline{\boldsymbol{V}}) \tag{3.8}$$

Where $\overline{\boldsymbol{V}}$ is the depth-averaged velocity given by $\overline{\boldsymbol{V}}(x,y,t) = \frac{1}{e}\int_{F_l}^{F_t} \boldsymbol{V} dz$ and $e(x,y,t)$ is the aperture. Second part of Equation (3.4) can be integrated as follows;

$$\int_{F_l}^{F_t} \boldsymbol{\nabla}.(\rho \boldsymbol{V}\boldsymbol{V})\, dz = \acute{\boldsymbol{\nabla}} \cdot \int_{F_l}^{F_t} (\rho \boldsymbol{V}\boldsymbol{V}) dz + \boldsymbol{\nabla} F_l \cdot \rho \boldsymbol{V}\boldsymbol{V}|_{f_l} - \boldsymbol{\nabla} F_t \cdot \rho \boldsymbol{V}\boldsymbol{V}|_{f_t} \tag{3.9}$$

Here $\acute{\boldsymbol{\nabla}} = \frac{\partial}{\partial \mathrm{x}}\underline{\mathbf{i}} + \frac{\partial}{\partial \mathrm{y}}\underline{\mathbf{j}}$ **,** and when no slip boundary conditions are assumed:

$$\int_{F_l}^{F_t} \boldsymbol{\nabla}.(\rho \boldsymbol{V}\boldsymbol{V})\, dz = \acute{\boldsymbol{\nabla}} \cdot (\rho e \overline{\boldsymbol{V}}\overline{\boldsymbol{V}}) \tag{3.10}$$

Taking $\boldsymbol{V}(x,y,z,t) = \overline{\boldsymbol{V}}(x,y,t) + \widetilde{\boldsymbol{V}}(x,y,z,t)$ where $\widetilde{\boldsymbol{V}}$ is the difference between the velocity and depth-averaged velocity, and its average over the aperture is zero, then,

$$\overline{\boldsymbol{V}}\overline{\boldsymbol{V}} = \frac{1}{e}\int_{F_l}^{F_t} \left(\overline{\boldsymbol{V}}\overline{\boldsymbol{V}} + 2\overline{\boldsymbol{V}}\widetilde{\boldsymbol{V}} + \widetilde{\boldsymbol{V}}\widetilde{\boldsymbol{V}}\right) dz$$

$$\overline{\boldsymbol{V}}\overline{\boldsymbol{V}} = \overline{\boldsymbol{V}}\overline{\boldsymbol{V}} + \overline{\widetilde{\boldsymbol{V}}\widetilde{\boldsymbol{V}}} \tag{3.11}$$

$$\int_{F_l}^{F_t} \boldsymbol{\nabla}.(\rho \boldsymbol{V}\boldsymbol{V})\, dz = \acute{\boldsymbol{\nabla}} \cdot (e\rho \overline{\boldsymbol{V}}\overline{\boldsymbol{V}}) + \acute{\boldsymbol{\nabla}} \cdot \left(e\rho \overline{\widetilde{\boldsymbol{V}}\widetilde{\boldsymbol{V}}}\right) \tag{3.12}$$

Integration of the third term of Equation (3.4) is given in the Equation (3.13).

$$\int_{F_l}^{F_t} \boldsymbol{\nabla}\mathrm{p}\, dz = e\acute{\boldsymbol{\nabla}}\mathrm{p} \tag{3.13}$$

The fourth term of Equation (3.4) is integrated as given below;

$$\int_{F_l}^{F_t} \rho \mathrm{g} \boldsymbol{\nabla}\acute{\mathrm{Z}}\, dz = \rho \mathrm{g} e \acute{\boldsymbol{\nabla}}\acute{\mathrm{Z}} \tag{3.14}$$

The integration of the fifth term of Equation (3.4) is outlined below,

$$\int_{F_l}^{F_t} \mu\nabla^2 \boldsymbol{V} dz = \mu \int_{F_l}^{F_t} (\boldsymbol{\nabla}\cdot\boldsymbol{\nabla})\boldsymbol{V} dz = \mu\left[\acute{\boldsymbol{\nabla}}\cdot\int_{F_l}^{F_t} \boldsymbol{\nabla V} dz + \boldsymbol{\nabla} F_l\cdot\boldsymbol{\nabla V}|_{f_l} - \boldsymbol{\nabla} F_t\cdot\boldsymbol{\nabla V}|_{f_t}\right] \quad (3.15)$$

$$\int_{F_l}^{F_t} \mu\nabla^2 \boldsymbol{V} dz = \mu\left[\acute{\boldsymbol{\nabla}}\cdot\left\{\acute{\boldsymbol{\nabla}}\int_{F_l}^{F_t} \boldsymbol{V} dz + \boldsymbol{\nabla} F_l\cdot\boldsymbol{V}|_{f_l} - \boldsymbol{\nabla} F_t\cdot\boldsymbol{V}|_{f_t}\right\} + \boldsymbol{\nabla} F_l\cdot\boldsymbol{\nabla V}|_{f_l} - \boldsymbol{\nabla} F_t\cdot\boldsymbol{\nabla V}|_{f_t}\right] \quad (3.16)$$

Taking the non-slip boundary conditions,

$$\int_{F_l}^{F_t} \mu\nabla^2 \boldsymbol{V} dz = \mu\acute{\nabla}^2(e\overline{\boldsymbol{V}}) + \mu\left\{\boldsymbol{\nabla} F_l\cdot\boldsymbol{\nabla V}|_{f_l} - \boldsymbol{\nabla} F_t\cdot\boldsymbol{\nabla V}|_{f_t}\right\} \quad (3.17)$$

Equations (3.8), (3.12), (3.13), (3.14) and (3.17) lead to Equation (3.18) using integration by parts, hence,.

$$\frac{\partial}{\partial t}(\rho e\overline{\boldsymbol{V}}) + \acute{\boldsymbol{\nabla}}\cdot(e\rho\overline{\boldsymbol{V}}\overline{\boldsymbol{V}}) + \acute{\boldsymbol{\nabla}}\cdot\left(e\rho\overline{\widetilde{\boldsymbol{V}}\widetilde{\boldsymbol{V}}}\right) + e\acute{\boldsymbol{\nabla}}\mathrm{p} - \rho g e\acute{\boldsymbol{\nabla}}\acute{\mathrm{Z}} - \mu\acute{\nabla}^2(e\overline{\boldsymbol{V}}) - \mu\left\{\boldsymbol{\nabla} F_l\cdot\boldsymbol{\nabla V}|_{f_l} - \boldsymbol{\nabla} F_t\cdot\boldsymbol{\nabla V}|_{f_t}\right\} = 0 \quad (3.18)$$

Having neglected the dispersive momentum flux $\rho\overline{\widetilde{\boldsymbol{V}}\widetilde{\boldsymbol{V}}}$ because it is much smaller than the advetive momentum flux $\rho\overline{\boldsymbol{V}}\overline{\boldsymbol{V}}$;

$$\frac{\partial}{\partial t}(\rho e\overline{\boldsymbol{V}}) + \acute{\boldsymbol{\nabla}}\cdot(e\rho\overline{\boldsymbol{V}}\overline{\boldsymbol{V}}) + e\acute{\boldsymbol{\nabla}}\mathrm{p} - \rho g e\acute{\boldsymbol{\nabla}}\acute{\mathrm{Z}} - \mu\acute{\nabla}^2(e\overline{\boldsymbol{V}}) - \mu\left\{\boldsymbol{\nabla} F_l\cdot\boldsymbol{\nabla V}|_{f_l} - \boldsymbol{\nabla} F_t\cdot\boldsymbol{\nabla V}|_{f_t}\right\} = 0 \quad (3.19)$$

Simplification of the term $\mu\left\{\boldsymbol{\nabla} F_l\cdot\boldsymbol{\nabla V}|_{f_l} - \boldsymbol{\nabla} F_t\cdot\boldsymbol{\nabla V}|_{f_t}\right\}$ in Equation (3.19) is explained below.

$$\boldsymbol{\nabla} F_l = \frac{\partial F_l}{\partial x}\boldsymbol{i} + \frac{\partial F_l}{\partial y}\boldsymbol{j} + \frac{\partial F_l}{\partial z}\boldsymbol{k} \quad (3.20)$$

The outer product of the divergence operator and velocity vector results in the following tensor.

$$\boldsymbol{\nabla V} = \begin{bmatrix} \frac{\partial u}{\partial x}\boldsymbol{ii} & \frac{\partial v}{\partial x}\boldsymbol{ij} & \frac{\partial w}{\partial x}\boldsymbol{ik} \\ \frac{\partial u}{\partial y}\boldsymbol{ji} & \frac{\partial v}{\partial y}\boldsymbol{jj} & \frac{\partial w}{\partial y}\boldsymbol{jk} \\ \frac{\partial u}{\partial z}\boldsymbol{ki} & \frac{\partial v}{\partial z}\boldsymbol{kj} & \frac{\partial w}{\partial z}\boldsymbol{kk} \end{bmatrix} \tag{3.21}$$

$\frac{\partial u}{\partial x} = \frac{\partial u}{\partial y} = \frac{\partial v}{\partial x} = \frac{\partial v}{\partial y} = \frac{\partial w}{\partial x} = \frac{\partial w}{\partial y} = 0$ when no-slip boundary conditions are assumed on the fracture surfaces, then the dot product of vector $\boldsymbol{\nabla} F_l$ and tensor $\boldsymbol{\nabla V}|_{f_l}$ becomes,

$$\boldsymbol{\nabla} F_l \cdot \boldsymbol{\nabla V}|_{f_l} = \frac{\partial F_l}{\partial z}\frac{\partial u}{\partial z}\Big|_{f_l}\boldsymbol{i} + \frac{\partial F_l}{\partial z}\frac{\partial v}{\partial z}\Big|_{f_l}\boldsymbol{j} + \frac{\partial F_l}{\partial z}\frac{\partial w}{\partial z}\Big|_{f_l}\boldsymbol{k} \tag{3.22}$$

When shearing of the joint walls is not considered their surface profiles can be given by the z coordinate of them.

$$F_l = z - z_l = 0 \tag{3.23}$$

$$F_t = z - z_l - e = 0 \tag{3.24}$$

The bottom wall was considered to be stationary with respect to the local coordinate system, and e is the height of fluid layer which did not change with z, therefore,

$$\frac{\partial F_l}{\partial z} = \frac{\partial F_t}{\partial z} = 1 \tag{3.25}$$

With laminar flow and no slip conditions at the boundaries, u or v were assumed to be have a parabolic distribution in the z direction, so the distribution of the velocity component 'u' in the z direction can be expressed as:

$$u = -\frac{6\bar{u}}{e^2}\left(z - \frac{e}{2}\right)\left(z + \frac{e}{2}\right) \tag{3.26}$$

Also,

$$\frac{\partial u}{\partial z} = -\frac{12\bar{u}}{e^2} z \tag{3.27}$$

Then Equation (3.19) can be rewritten as:

$$\frac{\partial}{\partial t}(\rho e\overline{\boldsymbol{V}}) + \acute{\boldsymbol{\nabla}} \cdot (e\rho\overline{\boldsymbol{V}}\overline{\boldsymbol{V}}) + e\acute{\boldsymbol{\nabla}}\mathrm{p} - \rho\mathrm{g}e\acute{\boldsymbol{\nabla}}\acute{\mathrm{Z}} - \mu\acute{\nabla}^2(e\overline{\boldsymbol{V}}) + \frac{12\mu}{e}\overline{\boldsymbol{V}} = 0 \tag{3.28}$$

Alternatively,

$$\frac{\partial}{\partial t}(\rho e\overline{\boldsymbol{V}}) = -\acute{\boldsymbol{\nabla}} \cdot (e\rho\overline{\boldsymbol{V}}\overline{\boldsymbol{V}}) + \mu\acute{\nabla}^2(e\overline{\boldsymbol{V}}) - \frac{12\mu}{e}\overline{\boldsymbol{V}} + \rho\mathrm{g}e\acute{\boldsymbol{\nabla}}\acute{\mathrm{Z}} - e\acute{\boldsymbol{\nabla}}\mathrm{p} \tag{3.29}$$

Equation (3.29) can be expressed in a scalar form as follows.

$$\frac{\partial}{\partial t}(\rho e\bar{u}) = -\frac{\partial}{\partial \mathrm{x}}(e\rho\bar{u}\bar{u}) - \frac{\partial}{\partial \mathrm{y}}(e\rho\bar{u}\bar{v}) + \mu\frac{\partial^2}{\partial \mathrm{x}^2}(e\bar{u}) + \mu\frac{\partial^2}{\partial \mathrm{y}^2}(e\bar{u}) - \frac{12\mu}{e}\bar{u} + \rho\mathrm{g}e\frac{\partial \acute{\mathrm{Z}}}{\partial \mathrm{x}} - e\frac{\partial \mathrm{p}}{\partial \mathrm{x}} \tag{3.30}$$

$$\frac{\partial}{\partial t}(\rho e\bar{v}) = -\frac{\partial}{\partial \mathrm{x}}(e\rho\bar{v}\bar{u}) - \frac{\partial}{\partial \mathrm{y}}(e\rho\bar{v}\bar{v}) + \mu\frac{\partial^2}{\partial \mathrm{x}^2}(e\bar{v}) + \mu\frac{\partial^2}{\partial \mathrm{y}^2}(e\bar{v}) - \frac{12\mu}{e}\bar{v} + \rho\mathrm{g}e\frac{\partial \acute{\mathrm{Z}}}{\partial \mathrm{y}} - e\frac{\partial \mathrm{p}}{\partial \mathrm{y}} \tag{3.31}$$

3.2.2 *Conservation of Mass*

The mass conservation of the flow is given by the continuity equation (3.32),

$$\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \boldsymbol{V}) = 0 \tag{3.32}$$

As described earlier, the continuity equation was also integrated along the z direction to result in an equivalent two-dimensional continuity equation.

$$\int_{F_l}^{F_t} \left(\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \boldsymbol{V}) \right) dz = 0 \tag{3.33}$$

Alternatively,

$$\frac{\partial(\rho e)}{\partial t} + \rho\left(\frac{\partial F_l}{\partial t} - \frac{\partial F_t}{\partial t}\right) + \acute{\nabla} \cdot (e\rho\bar{V}) = 0 \tag{3.34}$$

Since $F_t - F_l = e$,

$$\acute{\nabla} \cdot (\rho e\bar{V}) = -e\frac{\partial \rho}{\partial t} \tag{3.35}$$

Since the fluid is incompressible the result can be further reduced to:

$$\acute{\nabla} \cdot (\rho e\bar{V}) = 0 \tag{3.36}$$

When Equation (3.36) is expressed as a scalar equation,

$$\frac{\partial(\rho e\bar{u})}{\partial \mathrm{x}} + \frac{\partial(\rho e\bar{v})}{\partial \mathrm{y}} = 0 \tag{3.37}$$

Then Equation (3.30) can be expressed as follows:

$$\begin{aligned}\frac{\partial}{\partial t}(\rho e\bar{u}) = -(\rho e\bar{u})\frac{\partial \bar{u}}{\partial \mathrm{x}} - \bar{u}\frac{\partial(\rho e\bar{u})}{\partial \mathrm{x}} - (\rho e\bar{v})\frac{\partial \bar{u}}{\partial \mathrm{y}} - \bar{u}\frac{\partial(\rho e\bar{v})}{\partial \mathrm{y}} + \mu\frac{\partial^2}{\partial \mathrm{x}^2}(e\bar{u}) \\ + \mu\frac{\partial^2}{\partial \mathrm{y}^2}(e\bar{u}) - \frac{12\mu}{e}\bar{u} + \rho \mathrm{g} e\frac{\partial \acute{Z}}{\partial \mathrm{x}} - e\frac{\partial \mathrm{p}}{\partial \mathrm{x}}\end{aligned} \tag{3.38}$$

Substituting the mass conservation (Equation (3.37)) into the above momentum equation, following a further reduced momentum conservation equation, can be obtained for the velocity component '*u*'.

$$\begin{aligned}\frac{\partial}{\partial t}(\rho e\bar{u}) = -(\rho e\bar{u})\frac{\partial \bar{u}}{\partial \mathrm{x}} - (\rho e\bar{v})\frac{\partial \bar{u}}{\partial \mathrm{y}} + \mu\frac{\partial^2}{\partial \mathrm{x}^2}(e\bar{u}) + \mu\frac{\partial^2}{\partial \mathrm{y}^2}(e\bar{u}) - \frac{12\mu}{e}\bar{u} \\ + \rho \mathrm{g} e\frac{\partial \acute{Z}}{\partial \mathrm{x}} - e\frac{\partial \mathrm{p}}{\partial \mathrm{x}}\end{aligned} \tag{3.39}$$

By following the same procedure the following momentum conservation equation can be obtained for the velocity component '*v*'.

$$\frac{\partial}{\partial t}(\rho e\bar{v}) = -(\rho e\bar{u})\frac{\partial \bar{v}}{\partial \mathrm{x}} - (\rho e\bar{v})\frac{\partial \bar{v}}{\partial \mathrm{y}} + \mu\frac{\partial^2}{\partial \mathrm{x}^2}(e\bar{v}) + \mu\frac{\partial^2}{\partial \mathrm{y}^2}(e\bar{v}) - \frac{12\mu}{e}\bar{v} + \rho \mathrm{g} e\frac{\partial \acute{\mathrm{Z}}}{\partial \mathrm{y}} - e\frac{\partial \mathrm{p}}{\partial \mathrm{y}} \tag{3.40}$$

Equations (3.37), (3.39) and (3.40) are the governing equations of the proposed model. The aperture is a variable of 2D space and the irregular aperture is a parameter of the developed model. The model solutions are the velocity and pressure. The governing differential equations cannot be solved analytically, so a numerical solution has been suggested to calculate the velocity and pressure distribution of the flow domain.

3.3 Numerical Solution

Although the irregular aperture has been addressed in the flow model, the flow domain is still the plan view of the fracture, and therefore the governing equations can be descretised into a structured mesh created on the flow domain using the finite volume method. Each discrete volume contains three unknown variables, i.e. the velocity component '*u*', the velocity component '*v*', and the pressure and other known variables such as the aperture and fluid properties. The variables can be stored in a control volume in a collocated arrangement where all the variables are stored at the centre of the control volume. This method gives an error called odd-even decoupling when approximating differentiations such that when low order approximations (second order) are used, the value of the corresponding cell is not used in the differentiation assumption, which may lead to some errors. To remedy this problem Harlow and Welch (1965) suggested a staggered grid arrangement where the scalar variables (pressure, aperture, fluid density) are stored at the centre

of the control volume, the velocity component variables are stored at the faces of the control volume, and a staggered grid is used to assist with the nomenclature of the variables.

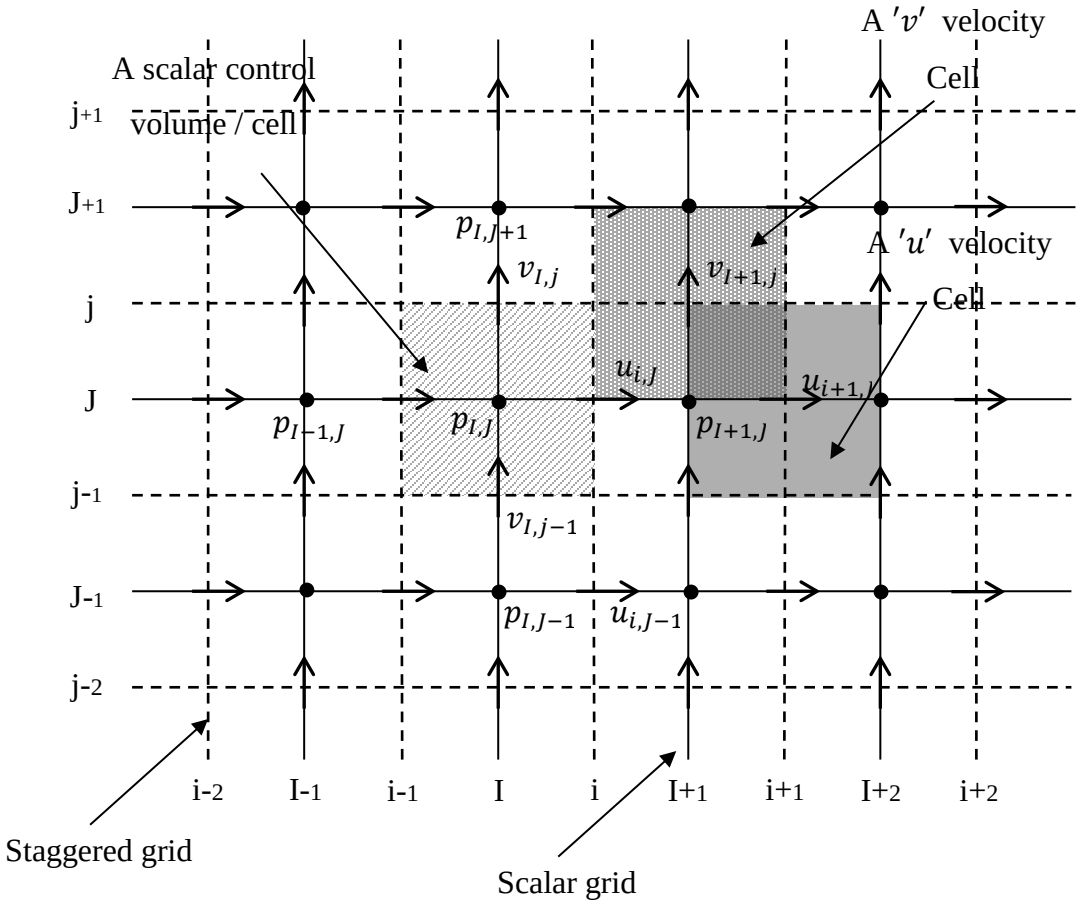


Figure 3-3: Staggered grid arrangement for velocity components

3.3.1 Staggered grid arrangement

The flow domain was descretised by a structured mesh shown as continuous lines in Figure 3-3. The X axis is numbered in capital '*I*'s and the Y axis in capital '*J*'s. The grid nodes are the centres of the control volumes (finite volumes), and the two velocity components '*u*' and '*v*' remain at the cell faces, as shown. A staggered grid

was laid in order to have staggered grid lines on the scalar cell faces which are numbered in simple 'i's in the X axis and simple 'j's in the Y axis. The 'u' and 'v' velocities of a particular cell, say (I,J) are numbered $u_{i,J}$ and $v_{I,j}$ respectively. A control volume (cell) for flow calculations and a cell for 'u' velocity and a cell for 'v' velocity are shown in Figure 3-3. The use of a staggered grid arrangement has benefits when coupling the pressure and velocity together.

The governing equations (i.e. Equations 3.37, 3.39 and 3.40) can be used for steady flow and transient flow calculations, although the transient term is omitted for steady flows. The numerical solution for steady flows is presented first.

3.4 Steady Flow

The advection term in the governing equation for the momentum conservation makes the equation non-linear. This component makes it impossible to solve the equation and the term has a negligible effect compared to the viscous terms for comparatively slow flows. Therefore, as is usual in numerical solutions to Navier-Stokes equations, the advection term has been omitted (Koyama 2007; Kishida et al. 2013; Zimmerman and Yeo 2013). As shown below, the governing Equations (3.13) and (3.14) have been simplified to solve steady flows numerically.

$$\mu\frac{\partial^2}{\partial \mathrm{x}^2}(eu)+\mu\frac{\partial^2}{\partial \mathrm{y}^2}(eu)-\frac{12\mu u}{e}+\rho \mathrm{g}e\frac{\partial \acute{\mathrm{Z}}}{\partial \mathrm{x}}-e\frac{\partial \mathrm{p}}{\partial \mathrm{x}}=0 \tag{3.41}$$

$$\mu\frac{\partial^2}{\partial \mathrm{x}^2}(ev)+\mu\frac{\partial^2}{\partial \mathrm{y}^2}(ev)-\frac{12\mu v}{e}+\rho \mathrm{g}e\frac{\partial \acute{\mathrm{Z}}}{\partial \mathrm{y}}-e\frac{\partial \mathrm{p}}{\partial \mathrm{y}}=0 \tag{3.42}$$

The velocity components in these equations are depth-averaged values and are shown without an over bar for simplicity.

3.4.1 Discretisation of the governing equations

Equation (3.41) has been integrated over the control volume, while the integration of Equation (3.41) for the 'u' control volume (i, J) is shown as parts in Equations (3.44) to (3.47).

$$\int_{cv}\left(\mu\frac{\partial^2}{\partial x^2}(eu)+\mu\frac{\partial^2}{\partial y^2}(eu)-\frac{12\mu u}{e}+\rho g e\frac{\partial \acute{Z}}{\partial x}-e\frac{\partial p}{\partial x}\right)dV=0 \tag{3.43}$$

$$\int_{cv}\left(\mu\frac{\partial^2}{\partial x^2}(eu)_{i,J}\right)dV=\mu\frac{\left[(eu)_{i-1,J}-2(eu)_{i,J}+(eu)_{i+1,J}\right]}{\Delta x}\Delta y \tag{3.44}$$

$$\int_{cv}\left(\mu\frac{\partial^2}{\partial y^2}(eu)_{i,J}\right)dV=\mu\frac{\left[(eu)_{i,J-1}-2(eu)_{i,J}+(eu)_{i,J+1}\right]}{\Delta y}\Delta x \tag{3.45}$$

$$\int_{cv}\left(-\frac{12\mu u}{e}+\rho g e\frac{\partial \acute{Z}}{\partial x}\right)_{i,J}dV=-\left(\frac{12\mu u}{e}\right)_{i,J}\Delta x\Delta y+(\rho g e)_{i,J}\Delta z\Delta y \tag{3.46}$$

$$\int_{cv}\left(-e\frac{\partial p}{\partial x}\right)_{i,J}dV=-\left(e_{(i,J)}\left(p_{I+1,J}-p_{I,J}\right)\right)\Delta y \tag{3.47}$$

The complete integrated equation is given by:

$$\mu\frac{\left[(eu)_{i-1,J}-2(eu)_{i,J}+(eu)_{i+1,J}\right]}{\Delta x^2}+\mu\frac{\left[(eu)_{i,J-1}-2(eu)_{i,J}+(eu)_{i,J+1}\right]}{\Delta y^2}$$
$$-\left(\frac{12\mu u}{e}\right)_{i,J}+(\rho g e)_{i,J}\frac{\Delta z}{\Delta x}-\left(e_{(i,J)}\frac{\left(p_{I+1,J}-p_{I,J}\right)}{\Delta x}\right)=0 \tag{3.48}$$

Following the same procedure for equation (3.42), by integrating over the 'v' control volume, (I, j), the following equation has been obtained.

$$\mu\frac{\left[(ev)_{I-1,j}-2(ev)_{I,j}+(ev)_{I+1,j}\right]}{\Delta x^2}+\mu\frac{\left[(ev)_{I,j-1}-2(ev)_{I,J}+(ev)_{I,j+1}\right]}{\Delta y^2}$$

$$-\left(\frac{12\mu v}{e}\right)_{I.j}+(\rho ge)_{I.j}\frac{\Delta z}{\Delta y}-\left(e_{(I,j)}\frac{\left(p_{I+1,J}-p_{I,J}\right)}{\Delta x}\right)=0 \tag{3.49}$$

The continuity equation for the flow domain is given by Equation (3.50), and it has been descretised using the finite volume method for the control volume (I,J) in the same manner to obtain the descretised continuity Equation (3.51) following the steps given below:

$$\frac{\partial(\rho eu)}{\partial x}+\frac{\partial(\rho ev)}{\partial y}=0 \tag{3.50}$$

$$\int_{cv}\left(\frac{\partial}{\partial x}(\rho eu)+\frac{\partial}{\partial x}(\rho ev)\right)dv=0$$

$$\left[(\rho eu)_{i,J}-(\rho eu)_{i-1,J}\right]\Delta y+\left[(\rho ev)_{I,j}-(\rho ev)_{I,j-1}\right]\Delta x=0$$

$$\frac{\left[(\rho eu)_{i,J}-(\rho eu)_{i-1,J}\right]}{\Delta x}+\frac{\left[(\rho ev)_{I,j}-(\rho ev)_{I,j-1}\right]}{\Delta y}=0 \tag{3.51}$$

The pressure only appears in the momentum conservation equations, because, there is no direct way to calculate pressure using these three equations so a pressure-velocity coupling technique must be used to achieve this. The techniques available for the Navier-Stokes equations can also be used for this model because the governing equations were derived from Navier-Stokes equations. Patanker and

Spalding (1972) developed the SIMPLE algorithm for this purpose and adopting the algorithm for the governing equations is explained below.

3.4.2 SIMPLE algorithm

SIMPLE stands for Semi Implicit Method for Pressure Linked Equations and it was developed by Patanker and Spalding (1972). Since we have momentum equations for the velocity components and continuity equation for the flow and no direct equation for pressure, this SIMPLE algorithm can be used to solve Navier-Stokes equations iteratively for velocities and pressure. The algorithm begins with initial values for pressure that are a best guess, and then the momentum equations are solved for velocities by substituting the guessed initial pressure. The continuity equation is then used to correct the pressure and then the corrected pressure is used to correct the velocity components. These new values are then taken to the next iteration and the same procedure is applied until the all corrections are negligible. A schematic representation of the SIMPLE algorithm is shown in Figure 3-4. The descretised momentum Equations (3.48) and (3.49) have been rearranged as follows, with explicit terms in RHS.

$$\begin{aligned}\mu\frac{\left[(eu^{\#})_{i-1,J}-2(eu^{\#})_{i,J}+(eu^{\#})_{i+1,J}\right]}{\Delta x^2}\\ +\mu\frac{\left[(eu^{\#})_{i,J-1}-2(eu^{\#})_{i,J}+(eu^{\#})_{i,J+1}\right]}{\Delta y^2}-\left(\frac{12\mu u^{\#}}{e}\right)_{i,J}\\ =-(\rho g e)_{i,J}\frac{\Delta z}{\Delta x}+\left(e_{(i,J)}\frac{\left(p^{\#}{}_{I+1,J}-p^{\#}{}_{I,J}\right)}{\Delta x}\right)=\varphi_{i,J}\end{aligned} \tag{3.52}$$

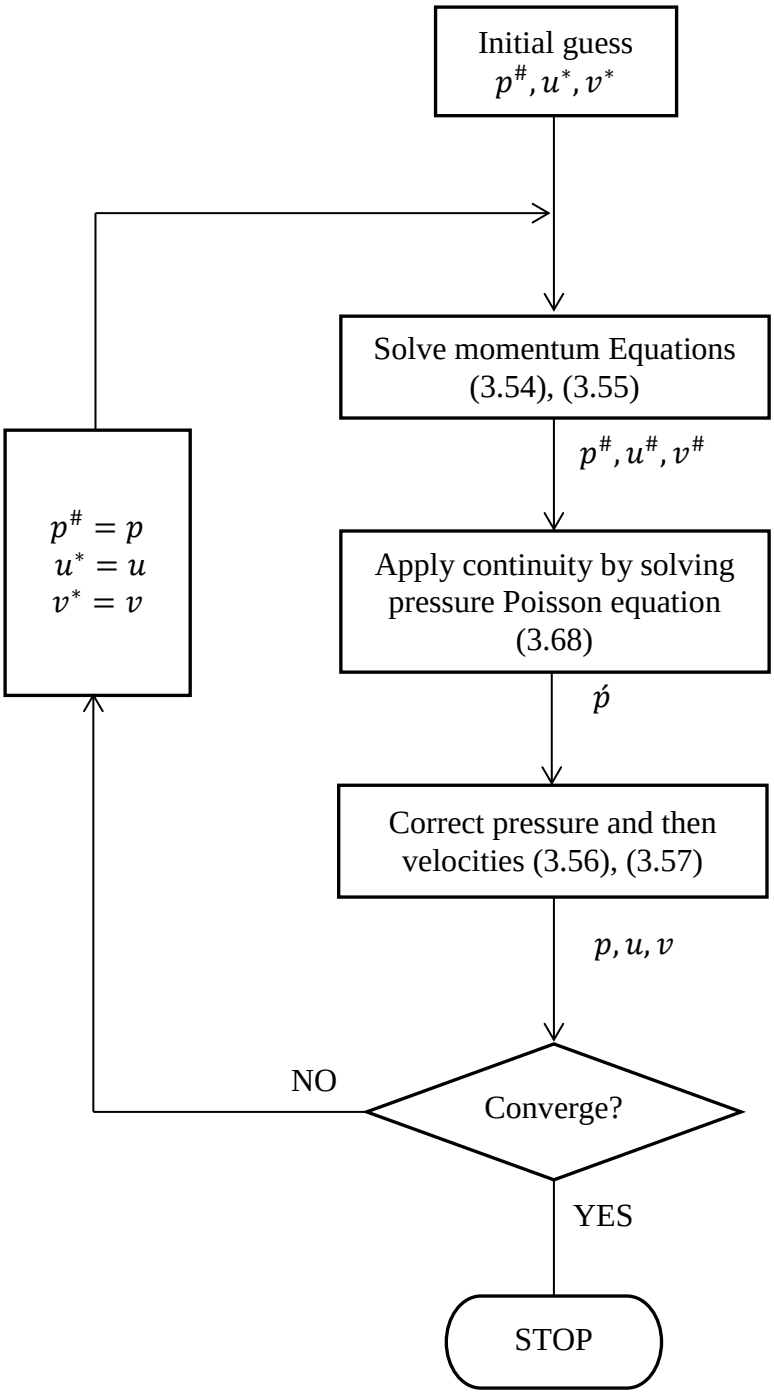


Figure 3-4: Flowchart of the SIMPLE algorithm

$$\mu\frac{\left[(ev^{\#})_{I-1,j}-2(ev^{\#})_{I,j}+(ev^{\#})_{I+1,j}\right]}{\Delta x^2}$$
$$+\mu\frac{\left[(ev^{\#})_{I,,j-1}-2(ev^{\#})_{I,j}+(ev^{\#})_{I,j+1}\right]}{\Delta y^2}-\left(\frac{12\mu v^{\#}}{e}\right)_{I,j} \tag{3.53}$$
$$=-(\rho ge)_{I.j}\frac{\Delta z}{\Delta y}+\left(e_{(I,j)}\frac{\left(p^{\#}{}_{I,J+1}-p^{\#}{}_{I,J}\right)}{\Delta y}\right)=\varphi_{I,j}$$

The velocities shown with a superscript '#' are the new intermediate velocities to be calculated. Now Equations (3.52) and (3.53) have been rearranged into Equations (3.54) and (3.55) respectively, in order to obtain descretised equations for the intermediate velocity calculation (i.e. $u^{\#}{}_{i,J}$ and $v^{\#}{}_{I,j}$) .

$$u^{\#}{}_{i,J}=-\frac{1}{B_{i,J}}\left[\frac{(\mu e)_{i-1,J}}{\Delta x^2}\right]u^{\#}{}_{i-1,J}-\frac{(\mu e)_{i+1,J}}{B_{i,J}\Delta x^2}u^{\#}{}_{i+1,J}$$
$$-\frac{1}{B_{i,J}}\left[\frac{(\mu e)_{i,J-1}}{\Delta y^2}\right]u^{\#}{}_{i,J-1}-\frac{1}{B_{i,J}}\left[\frac{(\mu e)_{i,J+1}}{\Delta y^2}\right]u^{\#}{}_{i,J+1} \tag{3.54}$$
$$-\frac{(\rho ge)_{i,J}}{B_{i,J}}\frac{\Delta z}{\Delta x}+\left(\frac{e_{(i,J)}}{B_{i,J}}\frac{\left(p^{\#}{}_{I+1,J}-p^{\#}{}_{I,J}\right)}{\Delta x}\right)$$

$$v^{\#}{}_{I,j}=-\frac{1}{B_{I,j}}\left[\frac{(\mu e)_{I-1,j}}{\Delta x^2}\right]v^{\#}{}_{I-1,j}-\frac{(\mu e)_{I+1,j}}{B_{I,j}\Delta x^2}v^{\#}{}_{I+1,j}$$
$$-\frac{1}{B_{I,j}}\left[\frac{(\mu e)_{I,,j-1}}{\Delta y^2}\right]v^{\#}{}_{I,j-1}-\frac{1}{B_{I,j}}\left[\frac{(\mu e)_{I,j+1}}{\Delta y^2}\right]v^{\#}{}_{I,j+1} \tag{3.55}$$
$$-\frac{(\rho ge)_{I.j}}{B_{I,j}}\frac{\Delta z}{\Delta y}+\left(\frac{e_{(I,j)}}{B_{I,j}}\frac{\left(p^{\#}{}_{I,J+1}-p^{\#}{}_{I,J}\right)}{\Delta y}\right)$$

In the above,

$$B_{i,J}=\left[-\frac{(2\mu e)_{i,J}}{\Delta x^2}-\frac{(2\mu e)_{i,J}}{\Delta y^2}-\left(\frac{12\mu}{e}\right)_{i,J}\right]$$

$$B_{I,j} = \left[-\frac{(2\mu e)_{I,j}}{\Delta x^2} - \frac{(2\mu e)_{I,j}}{\Delta y^2} - \left(\frac{12\mu}{e}\right)_{I,j} \right]$$

Having solved Equations (3.54) and (3.55), the intermediate velocity components are now calculated, but they only satisfy the momentum conservation, so the next step in the SIMPLE algorithm is to make them to satisfy mass conservation. By assuming that the intermediate values of $u^{\#}, v^{\#}, p^{\#}$ can be corrected with the corrections $\acute{u}, \acute{v}, \acute{p}$ respectively, if the corrected values are u, v, p, then the following relations are obtained.

$$u = u^{\#} + \acute{u} \tag{3.56}$$

$$v = v^{\#} + \acute{v} \tag{3.57}$$

$$p = p^{\#} + \acute{p} \tag{3.58}$$

Equation (3.54) obtained intermediate velocities, because, the pressure was an estimated field, but if the correct pressure that satisfies momentum and mass conservations were applied in Equation (3.54), the momentum conservation equation would result in the correct velocity, as follows;

$$\begin{aligned} u_{i,J} = & -\frac{1}{B_{i,J}}\left[\frac{(\mu e)_{i-1,J}}{\Delta x^2}\right] u_{i-1,J} - \frac{(\mu e)_{i+1,J}}{B_{i,J}\Delta x^2} u_{i+1,J} - \frac{1}{B_{i,J}}\left[\frac{(\mu e)_{i,J-1}}{\Delta y^2}\right] u_{i,J-1} \\ & - \frac{1}{B_{i,J}}\left[\frac{(\mu e)_{i,J+1}}{\Delta y^2}\right] u_{i,J+1} - \frac{(\rho g e)_{i,J}}{B_{i,J}}\frac{\Delta z}{\Delta x} \\ & + \left(\frac{e_{(i,J)}}{B_{i,J}}\frac{(p_{I+1,J} - p_{I,J})}{\Delta x}\right) \end{aligned} \tag{3.59}$$

By following operation (3.59) - (3.54) and substituting Equations (3.56) and (3.58), the following equation can be derived for the correction of velocity component $u_{i,J}$.

$$\acute{u}_{i,J} = -\frac{1}{B_{i,J}}\left[\frac{(\mu e)_{i-1,J}}{\Delta x^2}\right]\acute{u}_{i-1,J} - \frac{(\mu e)_{i+1,J}}{B_{i,J}\Delta x^2}\acute{u}_{i+1,J} - \frac{1}{B_{i,J}}\left[\frac{(\mu e)_{i,J-1}}{\Delta y^2}\right]\acute{u}_{i,J-1} - \frac{1}{B_{i,J}}\left[\frac{(\mu e)_{i,J+1}}{\Delta y^2}\right]\acute{u}_{i,J+1} + \left(\frac{e_{(i,J)}}{B_{i,J}}\frac{\left(\acute{p}_{I+1,J} - \acute{p}_{I,J}\right)}{\Delta x}\right) \tag{3.60}$$

Following the same procedure, Equation (3.61) is obtained for the correction of velocity component $v_{I,j}$.

$$\acute{v}_{I,j} = -\frac{(\mu e)_{I+1,j}}{B_{I,j}\Delta x^2}\acute{v}_{I+1,j} - \frac{1}{B_{I,j}}\left[\frac{(\mu e)_{I-1,j}}{\Delta x^2}\right]\acute{v}_{I-1,j} - \frac{1}{B_{I,j}}\left[\frac{(\mu e)_{I,j+1}}{\Delta y^2}\right]\acute{v}_{I,j+1} - \frac{1}{B_{I,j}}\left[\frac{(\mu e)_{I,,j-1}}{\Delta y^2}\right]\acute{v}_{I,j-1} + \left(\frac{e_{(I,j)}}{B_{I,j}}\frac{\left(\acute{p}_{I,J+1} - \acute{p}_{I,J}\right)}{\Delta y}\right) \tag{3.61}$$

All the velocity corrections of neighbouring cells in Equations (3.60) and (3.61) were omitted for simplicity and that is a typical approach in the SIMPLE algorithm. The assumption being made here is that the corrections were negligible when the solution was reached; then the velocity corrections become,

$$\acute{u}_{i,J} = \frac{e_{(i,J)}}{B_{i,J}}\left(\frac{\left(\acute{p}_{I+1,J} - \acute{p}_{I,J}\right)}{\Delta x}\right) \tag{3.62}$$

$$\acute{v}_{I,j} = \frac{e_{(I,j)}}{B_{I,j}}\left(\frac{\left(\acute{p}_{I,J+1} - \acute{p}_{I,J}\right)}{\Delta y}\right) \tag{3.63}$$

Equations (3.62) and (3.63) were used to correct the velocities by using corrections for pressure. The method for correcting the pressure is described below. Equations (3.62) and (3.63) were converted into corrected velocities by substituting Equations (3.56) and (3.57) respectively.

$$u_{i,J} = u^{\#}{}_{i,J} + \frac{e_{(i,J)}}{B_{i,J}}\left(\frac{\left(\acute{p}_{I+1,J} - \acute{p}_{I,J}\right)}{\Delta x}\right) \tag{3.64}$$

$$v_{I,j} = v^{\#}{}_{I,j} + \frac{e_{(I,j)}}{B_{I,j}}\left(\frac{\left(\acute{p}_{I,J+1} - \acute{p}_{I,J}\right)}{\Delta y}\right) \tag{3.65}$$

Now Equations (3.64) and (3.65) have been substituted to the discretised continuity Equation (3.51) to give:

$$\left[\frac{(\rho e_h u^{\#})_{i,J} + \frac{\rho e^2_{(i,J)}}{B_{i,J}}\left(\frac{\left(\acute{p}_{I+1,J} - \acute{p}_{I,J}\right)}{\Delta x}\right)}{\Delta \mathrm{x}} - \frac{(\rho e u^{\#})_{i-1,J} + \frac{\rho e^2_{(i-1,J)}}{B_{i-1,J}}\left(\frac{\left(\acute{p}_{I,J} - \acute{p}_{I-1,J}\right)}{\Delta x}\right)}{\Delta \mathrm{x}}\right] + \left[\frac{(\rho e v^{\#})_{I,j} + \frac{\rho e^2_{(I,j)}}{B_{I,j}}\left(\frac{\left(\acute{p}_{I,J+1} - \acute{p}_{I,J}\right)}{\Delta y}\right)}{\Delta y} - \frac{(\rho e v^{\#})_{I,j-1} + \frac{\rho e^2_{(I,j-1)}}{B_{I,j}}\left(\frac{\left(\acute{p}_{I,J} - \acute{p}_{I,J-1}\right)}{\Delta y}\right)}{\Delta y}\right] = 0 \tag{3.66}$$

The above can now be reduced to:

$$\left[\frac{(\rho eu^{\#})_{i,J} - (\rho eu^{\#})_{i-1,J}}{\Delta \mathrm{x}}\right] + \left[\frac{(\rho ev^{\#})_{I,j} - (\rho ev^{\#})_{I,j-1}}{\Delta y}\right]$$

$$= \frac{-\frac{\rho e^2_{(i,J)}}{B_{i,J}}\left(\frac{(\acute{p}_{I+1,J} - \acute{p}_{I,J})}{\Delta x}\right) + \frac{\rho e^2_{(i-1,J)}}{B_{i-1,J}}\left(\frac{(\acute{p}_{I,J} - \acute{p}_{I-1,J})}{\Delta x}\right)}{\Delta \mathrm{x}} \tag{3.67}$$

$$+ \frac{-\frac{\rho e^2_{(I,j)}}{B_{I,j}}\left(\frac{(\acute{p}_{I,J+1} - \acute{p}_{I,J})}{\Delta y}\right) + \frac{\rho e^2_{(I,j-1)}}{B_{I,j}}\left(\frac{(\acute{p}_{I,J} - \acute{p}_{I,J-1})}{\Delta y}\right)}{\Delta y}$$

Replacing the above with partial differential operators,

$$\frac{\partial}{\partial x}(\rho eu^{\#})_{I,J} + \frac{\partial}{\partial x}(\rho ev^{\#})_{I,J}$$

$$= M_{I,J}\acute{p}_{I,J} + M_{i,J}\acute{p}_{I+1,J} + M_{i-1,J}\acute{p}_{I-1,J} + M_{I,j}\acute{p}_{I,J+1} \tag{3.68}$$

$$+ M_{I,j-1}\acute{p}_{I,J-1}$$

Where,

$M_{i,J} = -\frac{\rho e^2_{(i,J)}}{B_{i,J}\Delta x^2}$, $M_{i-1,J} = -\frac{\rho e^2_{(i-1,J)}}{B_{i-1,J}\Delta x^2}$, $M_{I,j} = -\frac{\rho e^2_{(I,j)}}{B_{I,j}\Delta y^2}$, $M_{I,j-1} = -\frac{\rho e^2_{(I,j-1)}}{B_{I,j}\Delta y^2}$, and

$M_{I,J} = -M_{i,J} - M_{i-1,J} - M_{I,j} - M_{I,j-1}$

The pressure corrections were obtained by solving Equation (3.68). This equation is a Poisson type equation and is therefore called a Pressure Poisson equation. The pressure corrections were substituted into Equations (3.62) and (3.63) to obtain the corrected velocity components for the current cycle of iteration. When a correction is to be added to the pressure or velocity component, a suitable relaxation factor should be used to multiply it with; that will ensure convergence because the initial values are not always closer to the solution.

3.4.3 SIMPLEC algorithm

The omission made after Equations (3.60) and (3.61) makes the solution time considerably longer. The SIMPLEC (SIMPLE-Consistent) algorithm (Van Doormaal and Raithby 1984) was designed to reduce the computation time taken by the SIMPLE algorithm. Regardless of this omission, the corrections for the neighbouring cells were assumed to be equal to the central cell. Then the equations for the velocity component corrections can be modified to;

$$\acute{u}_{i,J} = \frac{e_{(i,J)}}{\overline{D}_{i.J}} \frac{\left(\acute{p}_{I+1,J} - \acute{p}_{I,J}\right)}{\Delta x} \tag{3.69}$$

$$\acute{v}_{i,J} = \frac{e_{(I,j)}}{\overline{D}_{I,J}} \frac{\left(\acute{p}_{I,J+1} - \acute{p}_{I,J}\right)}{\Delta y} \tag{3.70}$$

Where,

$$\overline{D}_{i.J} = \left[-\frac{(2\mu e)_{i,J}}{\Delta x^2} - \frac{(2\mu e)_{i,J}}{\Delta y^2} - \left(\frac{12\mu}{e}\right)_{i,J} + \frac{(\mu e)_{i+1,J}}{\Delta x^2} + \frac{(\mu e)_{i-1,J}}{\Delta x^2} + \frac{(\mu e)_{i,J+1}}{\Delta y^2} + \frac{(\mu e)_{i,J-1}}{\Delta y^2} \right]$$

$$\overline{D}_{I,j} = \left[-\frac{(2\mu e)_{I,j}}{\Delta x^2} - \frac{(2\mu e)_{I,j}}{\Delta y^2} - \left(\frac{12\mu}{e}\right)_{I,j} + \frac{(\mu e)_{I-1,j}}{\Delta x^2} + \frac{(\mu e)_{I+1,j}}{\Delta x^2} + \frac{(\mu e)_{I,j+1}}{\Delta y^2} + \frac{(\mu e)_{I,,j-1}}{\Delta y^2} \right]$$

Thereafter, by following the same procedure, the pressure Poisson equation for the pressure corrections was obtained as follows.

$$\begin{aligned} \frac{\partial}{\partial x}(\rho e u^{\#})_{I,J} + \frac{\partial}{\partial x}(\rho e v^{\#})_{I,J} \\ = M_{I,J}\acute{p}_{I,J} + M_{i,J}\acute{p}_{I+1,J} + M_{i-1,J}\acute{p}_{I-1,J} + M_{I,j}\acute{p}_{I,J+1} \\ + M_{I,j-1}\acute{p}_{I,J-1} \end{aligned} \tag{3.71}$$

Where,

$M_{i,J} = -\frac{\rho e^2_{(i,J)}}{\bar{D}_{i,J}\Delta x^2}$, $M_{i-1,J} = -\frac{\rho e^2_{(i-1,J)}}{\bar{D}_{i-1,J}\Delta x^2}$, $M_{I,j} = -\frac{\rho e^2_{(I,j)}}{\bar{D}_{I,j}\Delta y^2}$, $M_{I,j-1} = -\frac{\rho e^2_{(I,j-1)}}{\bar{D}_{I,j-1}\Delta y^2}$, and

$M_{I,J} = -M_{i,J} - M_{i-1,J} - M_{I,j} - M_{I,j-1}$

Following the same procedure as the SIMPLE algorithm, corrections were obtained and iterated until a solution with negligible corrections was reached. SIMPLEC can handle higher pressure relaxations to give faster solutions, whereas SIMPLE is slow with higher relaxations and might diverge for some higher relaxation factors, while SIMPLEC still can provide convergence.

3.5 Transient Flow

The effective normal stress applied to the fracture deforms the fracture mechanically. This effect can be coupled with time according to the behaviour of the stress environment to simulate the flow behaviour changes upon time.

3.5.1 *Discretising the Governing Equations*

When it comes to transient flow, a finite volume should be considered in a finite time element because the flow data of a particular control volume changes over time. To achieve this result, the governing equations (3.30), (3.31), and (3.37) should be integrated over a control volume and over a small time interval.

3.5.1.1 Momentum equations

$$\int_{t}^{t+\Delta t}\int_{cv}\left[\frac{\partial}{\partial t}(\rho eu)\right]dvdt = \int_{t}^{t+\Delta t}\int_{cv}\left[+\mu\frac{\partial^2}{\partial x^2}(eu)+\mu\frac{\partial^2}{\partial y^2}(eu)-\frac{12\mu}{e}u+\rho ge\frac{\partial \acute{Z}}{\partial x}-e\frac{\partial p}{\partial x}\right]dvdt \tag{3.72}$$

The next step in integrating gives:

$$\int_{t}^{t+\Delta t}\int_{cv}\left[\frac{\partial}{\partial t}(\rho ev)\right]dvdt = \int_{t}^{t+\Delta t}\int_{cv}\left[+\mu\frac{\partial^2}{\partial x^2}(ev)+\mu\frac{\partial^2}{\partial y^2}(ev)-\frac{12\mu}{e}v+\rho ge\frac{\partial \acute{Z}}{\partial y}-e\frac{\partial p}{\partial y}\right]dvdt \tag{3.73}$$

These equations were integrated numerically over the ‘u’ velocity cells and ‘v’ velocity cells accordingly, and over a small time interval, thus for $u_{i,J}$ term,

$$\frac{(\rho eu)_{i,J}^{n+1}-(\rho eu)_{i,J}^{n}}{\Delta t} = \mu\frac{\left[(eu)_{i-1,J}^{n+1}-2(eu)_{i,J}^{n+1}+(eu)_{i+1,J}^{n+1}\right]}{\Delta x^2} + \mu\frac{\left[(eu)_{i,J-1}^{n+1}-2(eu)_{i,J}^{n+1}+(eu)_{i,J+1}^{n+1}\right]}{\Delta y^2}-\frac{12\mu}{e}u_{i,J}^{n+1} + (\rho ge)_{i,J}^{n}\frac{\Delta z}{\Delta x}-\left(e\frac{\Delta p}{\Delta x}\right)_{i,J}^{n+1} \tag{3.74}$$

For $v_{I,j}$ component,

$$\begin{aligned}\frac{(\rho ev)_{I,j}^{n+1}-(\rho ev)_{i,j}^{n}}{\Delta t} &= \mu\frac{\left[(ev)_{I-1,j}^{n+1}-2(ev)_{I,j}^{n+1}+(ev)_{I+1,j}^{n+1}\right]}{\Delta x^2} \\ &+ \mu\frac{\left[(ev)_{I,j-1}^{n+1}-2(ev)_{I,j}^{n+1}+(ev)_{I,j+1}^{n+1}\right]}{\Delta y^2}-\frac{12\mu}{e}v_{I,j}^{n+1} \\ &+ (\rho ge)_{I,j}^{n}\frac{\Delta z}{\Delta y}-\left(e\frac{\Delta p}{\Delta y}\right)_{I,j}^{n+1}\end{aligned} \tag{3.75}$$

3.5.1.2 Continuity equation

$$\int_{t}^{t+\Delta t}\int_{cv}\left(\frac{\partial}{\partial \mathrm{x}}(\rho eu)+\frac{\partial}{\partial \mathrm{x}}(\rho ev)\right)dvdt=0 \tag{3.76}$$

The descretised continuity Equation (3.77) can be obtained after numerically integrating Equation (3.76) over a scalar control volume and over a small time interval, hence,

$$\frac{\left[(\rho eu)_{i,J}^{n+1}-(\rho eu)_{i-1,J}^{n+1}\right]}{\Delta x}+\frac{\left[(\rho ev)_{I,j}^{n+1}-(\rho ev)_{I,j-1}^{n+1}\right]}{\Delta y}=0 \tag{3.77}$$

3.5.2 SIMPLE transient algorithm

SIMPLE or SIMPLEC algorithms were used to solve the pressure and velocities inside a time step, and once a stable solution is sought, the time step is advanced. For the inner iterations in transient calculations, Equations (3.74) and (3.75) were rearranged with explicit terms in RHS. Inside the inner iteration loop, the terms in

the time interval $n+1$ are shown with a superscript 'm' to show that the values are still being updated to reach the $n+1$ time interval.

$$\frac{(\rho eu)_{i,J}^{m\#}}{\Delta t} - \frac{(\mu eu)_{i-1,J}^{m\#}}{\Delta x^2} + \frac{2(\mu eu)_{i,J}^{m\#}}{\Delta x^2} - \frac{(\mu eu)_{i+1,J}^{m\#}}{\Delta x^2} - \frac{(\mu eu)_{i,J-1}^{m\#}}{\Delta y^2} + \frac{2(\mu eu)_{i,J}^{m\#}}{\Delta y^2} - \frac{(\mu eu)_{i,J+1}^{m\#}}{\Delta y^2} + \frac{12\mu}{e} u_{i,J}^{m\#} = \frac{(\rho eu)_{i,J}^{n}}{\Delta t} + (\rho ge)_{i,J}^{n} \frac{\Delta z}{\Delta x} - \left(e \frac{\Delta p}{\Delta x}\right)_{i,J}^{m\#} = \phi_{i,J} \tag{3.78}$$

$$\frac{(\rho ev)_{I,j}^{m\#}}{\Delta t} - \frac{(\mu ev)_{I-1,j}^{m\#}}{\Delta x^2} + \frac{2(\mu ev)_{I,j}^{m\#}}{\Delta x^2} - \frac{(\mu ev)_{I+1,j}^{m\#}}{\Delta x^2} - \frac{(\mu ev)_{I,j-1}^{m\#}}{\Delta y^2} + \frac{2(\mu ev)_{I,j}^{m\#}}{\Delta y^2} - \frac{(\mu ev)_{I,j+1}^{m\#}}{\Delta y^2} + \frac{12\mu}{e} v_{I,j}^{m\#} = \frac{(\rho ev)_{i,j}^{n}}{\Delta t} + (\rho ge)_{I,j}^{n} \frac{\Delta z}{\Delta y} - \left(e \frac{\Delta p}{\Delta y}\right)_{I,j}^{m\#} = \phi_{I,j} \tag{3.79}$$

Alternatively,

$$e_{(i,J)} \left[\frac{\rho}{\Delta t} + \frac{2\mu}{\Delta x^2} + \frac{2\mu}{\Delta y^2} + \frac{12\mu}{e^2}\right] u_{i,J}^{m\#} - e_{(i-1,J)} \frac{\mu}{\Delta x^2} u_{i-1,J}^{m\#} - e_{h_{(i+1,J)}} \frac{\mu}{\Delta x^2} u_{i+1,J}^{m\#} - e_{(i,J-1)} \frac{\mu}{\Delta y^2} u_{i,J-1}^{m\#} - e_{(i,J+1)} \frac{\mu}{\Delta y^2} u_{i,J+1}^{m\#} = \phi_{i,J} \tag{3.80}$$

For the v component,

$$e_{(I,j)} \left[\frac{\rho}{\Delta t} + \frac{2\mu}{\Delta x^2} + \frac{2\mu}{\Delta y^2} + \frac{12\mu}{e^2}\right] v_{I,j}^{m\#} - e_{(I-1,j)} \frac{\mu}{\Delta x^2} v_{I-1,j}^{m\#} - e_{(I+1,j)} \frac{\mu}{\Delta x^2} v_{I+1,j}^{m\#} - e_{(I,j-1)} \frac{\mu}{\Delta y^2} v_{I,j-1}^{m\#} - e_{(I,j+1)} \frac{\mu}{\Delta y^2} v_{I,j+1}^{m\#} = \phi_{I,j} \tag{3.81}$$

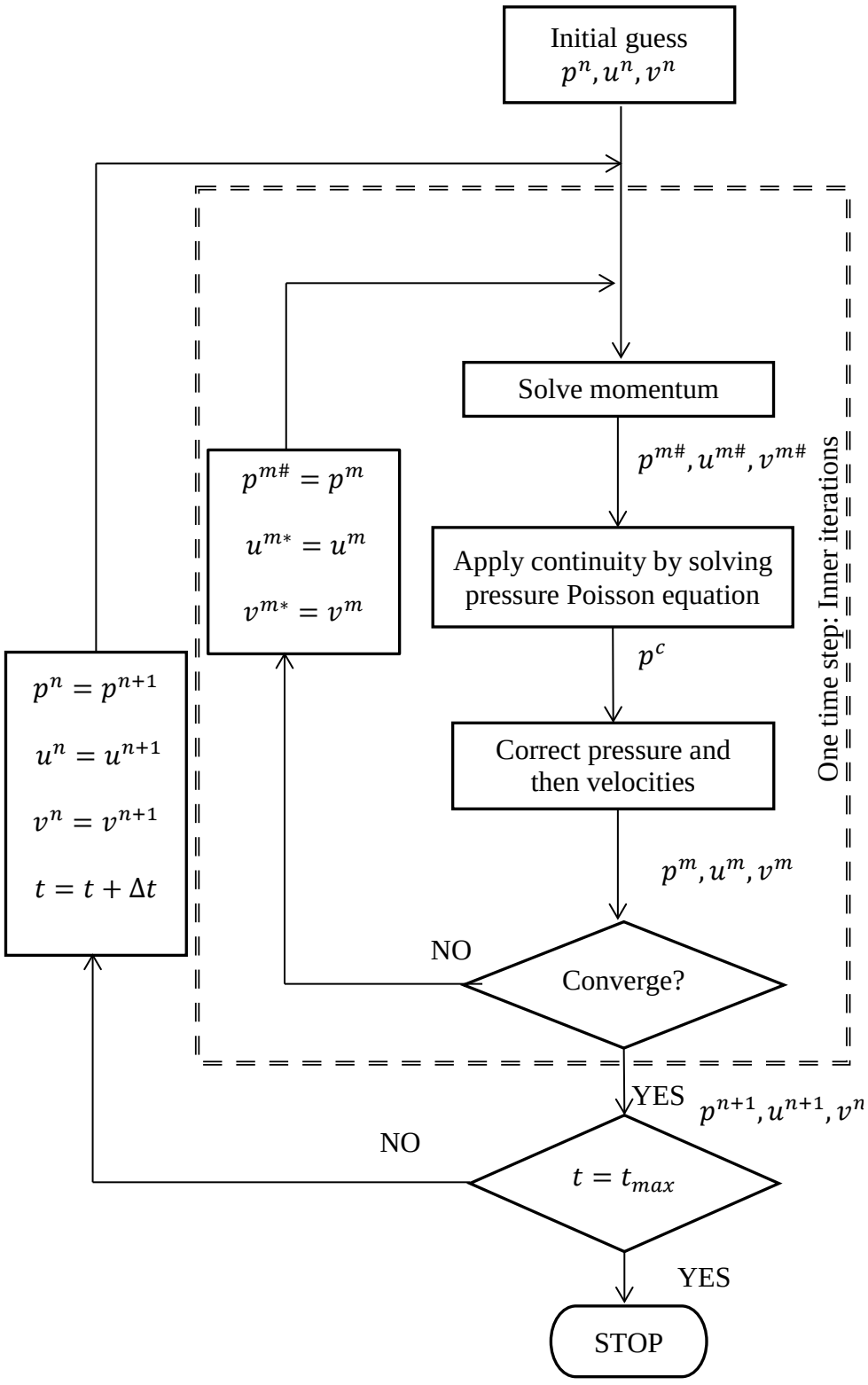


Figure 3-6: Transient SIMPLE algorithm flow chart

Inner iterations began by solving Equations (3.80) and (3.81) for intermediate velocities. The same procedure described in the SIMPLE/SIMPLEC method was followed to correct the velocities and pressure. The corrections for the velocities will be;

$$u_{i,J}^{n+1} = u_{i,J}^{m\#} + \frac{1}{D_{i,J}} \frac{\left(\acute{p}_{I+1,J} - \acute{p}_{I,J}\right)}{\Delta x} \tag{3.82}$$

$$v_{I,j}^{n+1} = v_{I,j}^{m\#} + \frac{1}{D_{I,j}} \frac{\left(\acute{p}_{I,J+1} - \acute{p}_{I,J}\right)}{\Delta y} \tag{3.83}$$

In the above,

$$D_{i,J} = \left[-\frac{\rho}{\Delta t} - \frac{2\mu}{\Delta x^2} - \frac{2\mu}{\Delta y^2} - \frac{12\mu}{e_{(i,J)}{}^2} \right]$$

$$D_{I,j} = \left[-\frac{\rho}{\Delta t} - \frac{2\mu}{\Delta x^2} - \frac{2\mu}{\Delta y^2} - \frac{12\mu}{e_{(I,j)}{}^2} \right]$$

Corrected velocities were substituted into the descretised continuity equation to form the pressure Poisson equation. The flow chart for the transient SIMPLE algorithm is shown in Figure 3-6.

$$\frac{\left[(\rho e u)_{i,J}^{m\#} + \left(\frac{\rho e}{D_{i,J}}\right)^{n+1} \frac{\left(\acute{p}_{I+1,J} - \acute{p}_{I,J}\right)}{\Delta x} - (\rho e u)_{i-1,J}^{m\#} - \left(\frac{\rho e}{D_{i-1,J}}\right)^{n+1} \frac{\left(\acute{p}_{I,J} - \acute{p}_{I-1,J}\right)}{\Delta x} \right]}{\Delta x}$$

$$+ \frac{\left[(\rho e v)_{I,j}^{m\#} + \left(\frac{\rho e}{D_{I,j}}\right)^{n+1} \frac{\left(\acute{p}_{I,J+1} - \acute{p}_{I,J}\right)}{\Delta y} - (\rho e v)_{I,j-1}^{n+1} - \left(\frac{\rho e}{D_{I,j-1}}\right)^{n+1} \frac{\left(\acute{p}_{I,J} - \acute{p}_{I,J-1}\right)}{\Delta y} \right]}{\Delta y} \tag{3.84}$$

$$= 0$$

Alternatively,

$$\left(\frac{\rho e}{D_{i,J}\Delta x^2}\right)^{n+1} \acute{p}_{I+1,J} - \left(\frac{\rho e}{D_{i,J}\Delta x^2}\right)^{n+1} \acute{p}_{I,J} - \left(\frac{\rho e}{D_{i-1,J}\Delta x^2}\right)^{n+1} \acute{p}_{I,J}$$
$$+ \left(\frac{\rho e}{D_{i-1,J}\Delta x^2}\right)^{n+1} \acute{p}_{I-1,J} + \left(\frac{\rho e}{D_{I,j}\Delta y^2}\right)^{n+1} \acute{p}_{I,J+1}$$
$$- \left(\frac{\rho e}{D_{I,j}\Delta y^2}\right)^{n+1} \acute{p}_{I,J} - \left(\frac{\rho e}{D_{I,j-1}\Delta y^2}\right)^{n+1} \acute{p}_{I,J} \tag{3.85}$$
$$+ \left(\frac{\rho e}{D_{I,j-1}\Delta y^2}\right)^{n+1} \acute{p}_{I,J-1}$$
$$= -\frac{(\rho eu)_{i,J}^{m\#} - (\rho eu)_{i-1,J}^{m\#}}{\Delta x} - \frac{(\rho ev)_{I,j}^{m\#} - (\rho ev)_{I,j-1}^{n+1}}{\Delta y}$$

Simplifying the above leads to:

$$H_{I,J}\acute{p}_{I,J} + H_{I+1,J}\acute{p}_{I+1,J} + H_{I-1,J}\acute{p}_{I-1,J} + H_{I,J+1}\acute{p}_{I,J+1} + H_{I,J-1}\acute{p}_{I,J-1}$$
$$= -\frac{(\rho eu)_{i,J}^{m\#} - (\rho eu)_{i-1,J}^{m\#}}{\Delta x} - \frac{(\rho ev)_{I,j}^{m\#} - (\rho ev)_{I,j-1}^{n+1}}{\Delta y} \tag{3.86}$$

Where,

$H_{I+1,J} = \left(\frac{\rho e}{D_{i,J}\Delta x^2}\right)^{n+1} \acute{p}_{I+1,J}$, $H_{I-1,J} = \left(\frac{\rho e}{D_{i-1,J}\Delta x^2}\right)^{n+1}$, $H_{I,J+1} = \left(\frac{\rho e}{D_{I,j}\Delta y^2}\right)^{n+1}$,

$H_{I,J-1} = \left(\frac{\rho e}{D_{I,j-1}\Delta y^2}\right)^{n+1}$, and $H_{I-1,J} = -H_{I-1,J} - H_{I+1,J} - H_{I,J-1} - H_{I,J+1}$

3.5.3 *SIMPLEC Transient Algorithm*

For transient SIMPLEC method too, Equations (3.80) and (3.81) were used to calculate the intermediate velocities. Once the intermediate velocities in a particular time step have been calculated, then the velocity and pressure correction equations can be derived. In SIMPLEC the neighbouring cell velocity corrections were assumed to be equal to the corrected middle cell velocities, and then the velocity and pressure corrections are related as follows;

$$\acute{u}_{i,J} = \frac{\rho e_{(i,J)}}{D_{i.J}} \frac{\left(\acute{p}_{I+1,J} - \acute{p}_{I,J}\right)}{\Delta x} \tag{3.87}$$

$$\acute{v}_{i,J} = \frac{\rho e_{(I,j)}}{D_{I,J}} \frac{\left(\acute{p}_{I,J+1} - \acute{p}_{I,J}\right)}{\Delta y} \tag{3.88}$$

Where,

$$D_{i.J} = -\frac{\rho}{\Delta t} - \frac{12\mu}{e_{(i,J)}} + \left(e_{(i-1,J)} + e_{(i+1,J)} - 2e_{(i,J)}\right)\left[\frac{\mu}{\Delta x^2}\right] + \left(e_{(i,J-1)} + e_{(i,J+1)} - 2e_{(i,J)}\right)\left[\frac{\mu}{\Delta y^2}\right] \tag{3.89}$$

$$D_{I,j} = -\frac{\rho}{\Delta t} - \frac{12\mu}{e_{(I,j)}} + \left(e_{(I-1,j)} + e_{(I+1,j)} - 2e_{(I,j)}\right)\left[\frac{\mu}{\Delta x^2}\right] + \left(e_{(I,j-1)} + e_{(I,j+1)} - 2e_{(I,j)}\right)\left[\frac{\mu}{\Delta y^2}\right] \tag{3.90}$$

The pressure correction equations were the same as the SIMPLE transient method apart from expressions $D_{i.J}$ and $D_{I,j}$ which should be calculated using Equations (3.89) and (3.90) respectively.

3.6 Fracture deformation

The mechanical deformation of a rock fracture upon an applied normal stress is given by the following relationship (Kulhawy 1975; Bandis et al. 1983).

$$\Delta v = \frac{\acute{\sigma_n} v_m}{K_i v_m + \acute{\sigma_n}} \tag{3.91}$$

Δv is normal deformation at the effective normal stress $\acute{\sigma_n}$ with respect to the initial effective normal stress. v_m is maximum closure and K_i is the initial joint normal stiffness. When this relationship has been applied to a fracture, a common maximum closure should be chosen. Some empirical relations were proposed in order to

calculate the maximum closure if the fracture has a higher variance in the aperture distribution, using an average maximum closure did not give promising results from the above relationship. It was therefore proposed that this relationship would work well for mated joints. If a non-mated joint is considered to be in two dimensions, every aperture in the fracture can be treated individually. Then their maximum closures will be their own initial apertures. Having digitised Equation (3.91) into Equation (3.92) in a 2D finite volume domain, the deformation of a non-mated joint can then be calculated.

$$\Delta v_{[I,J]} = \frac{\acute{\sigma}_{n[I,J]} v_{m[I,J]}}{K_i v_{m[I,J]} + \acute{\sigma}_{n[I,J]}} \tag{3.92}$$

Equation (3.92) can now be used to calculate the normal deformations of the aperture when the confining pressure has increased in steady flow calculations, or when calculations for the next time step in transient flow calculations have commenced.

3.7 Summary

In this chapter, the development of a mathematical model and numerical solution for a single phase flow through rough rock fracture were presented. The model development began by integrating the three-dimensional Navier-Stokes equation in the direction of the fracture aperture, while considering the deformable fracture walls as the limits of integration. That resulted in an equivalent two-dimensional Navier-Stokes equation with a fracture aperture included as a variable in 2D space and time. This developed model is a non-linear partial differential equation which cannot be

solved explicitly due to non-linearity and insufficient governing equations for the velocity components and fluid pressure. Therefore the model was solved numerically by discretising the governing equations in the domain using the Finite Volume Method and using the SIMPLE algorithm to couple the velocity and pressure. For faster convergence, the SIMPLE algorithm modification to the SIMPLEC algorithm was also presented. The method for solving steady flows and transient flows were derived separately in this chapter. To obtain a solution, a computer programme is required and its development is described in the following chapter.

Chapter 4 Rock Fracture Flow Solver (RFFS)

4.1 Introduction

A mathematical model for flow through a rough deformable rock fracture was proposed in Chapter 3. Velocity and pressure constitute the main solution components of the model, and because it is a non-linear partial differential equation, a numerical solution was suggested in Chapter 3 requiring a computer program to facilitate solution. This chapter outlines the development of this computer programme, RFFS: Rock Fracture Flow Solver using the MATLAB programming language.

MATLAB stands for 'matrix laboratory', and its programming language is a powerful tool to perform mathematical calculations that deal with the matrices that include of unknown variables of the mathematical model. MATLAB consists of a graphical user interface design environment (GUIDE) that designs the graphical user interfaces (GUI) which is the communicator between the end user and RFFS. GUIDE provides more benefits when using MATLAB to develop a computer programme to make it user-friendly.

How the programme structure and the subroutines (functions) were developed is described first, followed by a description of the design and arrangement of the graphical user interfaces. The function codes are given in Appendix A.

4.2 Development of the Flow Domain

The first step in the calculation is to develop the flow domain where the user provides the necessary domain details such as the length and width of the rock fracture and the required discretisation dimensions (the number of control volumes in one direction). According to the required discretisation, the size of the matrix of each variable is determined according to the staggered grid arrangement suggested in Chapter 3. If the number of cells in the X-direction is *nx* and *ny* in the Y-direction, the sizes of the variable matrices are given in Table 3.1. The matrix rows are aligned in the X-direction of the domain and the columns in the Y-direction. Typical storage positions for the unknown variables are shown in Figure 3.7 (Scalar grids are shown as continuous lines and broken lines indicate the staggered grids). Aperture variables are stored at the same place the pressure was stored for a particular control volume. All other properties such as fluid density, dynamic viscosity of fluid are also stored at the centre of the cell. However, for an incompressible, homogenous fluid these properties can be taken as constants, and positioning at the control volume will be unnecessary.

Table 3.1: Matrix sizes of the variables

Variable	*Grid arrangement*	*Rows*	*Columns*
Velocity component ‘u’ (U)	Staggered	nx+1	ny
Velocity component ‘v’ (V)	Staggered	nx+2	ny-1
Pressure (P)	Scalar	nx	ny
Aperture (e)	Scalar	nx	ny

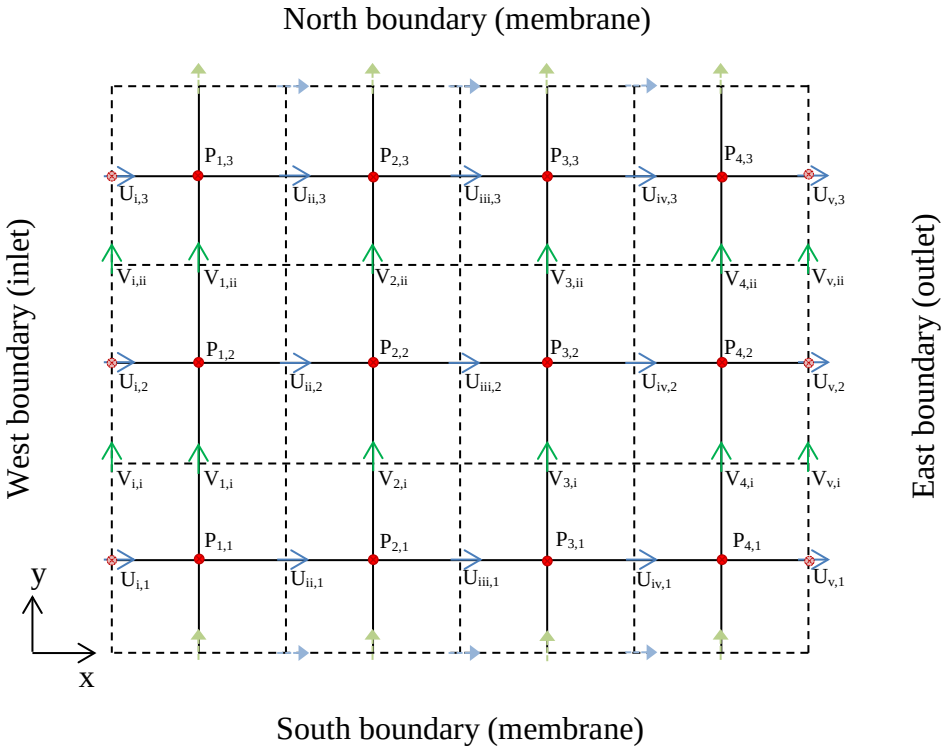


Figure 3.1: Unknown variable storage positions on a sample flow domain with $nx = 4$ and $ny = 3$

4.2.1 *Boundary Values*

The boundary values of the variables are determined according to the calculation procedure, while the inlet and outlet velocity boundary values are calculated with the

domain velocity calculations. This means that those boundary variables are also included in the main velocity variable matrix. The side boundary (north and south boundaries of Figure 3.7) velocity variables are calculated according to the boundary condition, so they are stored in separate column vectors and values are assigned to them accordingly. If the boundary is considered to be a wall in the triaxial test, then the velocities are always zero to assume no slip boundaries (Dirichlet boundary condition). When real conditions are assumed, the north and south boundaries of Figure 3.7 become symmetrical boundaries, and in those conditions the north and south velocity vectors are updated at each time domain, and the velocities are updated by assuming that the velocity gradient at the boundary is zero (Neumann boundary condition). The inlet and outlet pressures are fixed (Dirichlet boundary condition) and are assigned to two separate row vectors for further use in the calculations. The normal pressure gradient at the north and south boundaries was assumed to be vanished ($\nabla P . \boldsymbol{n} = 0,$ where $\boldsymbol{n}$ is the unit normal vector at the boundary). The aperture boundary values are calculated by extrapolating the domain values.

4.2.2 *Initial Values*

All the velocities are assumed to be zero initially and pressure is considered to be uniformly distributed according to the inlet and outlet pressures given by the user. The user input for the aperture distribution is a matrix of aperture heights in 2D space that may contain the aperture data according to the resolution of the scanned data. The user preference of domain discretisation may not be the same as the scan data discretisation, so a function was used to create the initial aperture matrix, according to the user preference discretisation, using an area weighted interpolation method

(*descret( ),* Appendix A). This aperture data was recorded without any normal stress being applied so this initial aperture matrix was used as the maximum deformation matrix of the aperture. Thereafter, the initial deformation was calculated according to the initial confining stress and initial joint normal stiffness given by the user. The aperture matrix was updated after the initial deformation was applied, and then used for the calculation.

4.3 Calculation

To perform the calculations, all the variables must be combined with the boundary values, so the updated variables matrices were combined with the updated boundary vectors to create the matrices with boundary values (Figure 3.8). The four corner elements are totally inactive and were never used in any calculation, but to create the total matrix some value must be assigned to them, so an adjacent boundary value was copied to those four elements. With the aperture contacts, those velocities which should have been zero because of the local boundaries, were also calculated and updated as described in Section 4.5. The model has terms where the velocities are multiplied with the aperture. To calculate those terms, new aperture matrices were made where the velocities are, because originally they were at the centre of the cell and velocities are kept at the faces of the cell. An averaging function '*center2uv(e)*' (Appendix A) return two matrices of '*eu*' and '*ev*' those contain the aperture at the cell faces at the 'U' velocity positions and 'V' velocity positions. Now these two matrices have been updated with the apertures at the boundary values of the vectors and created the final matrices for the calculations.

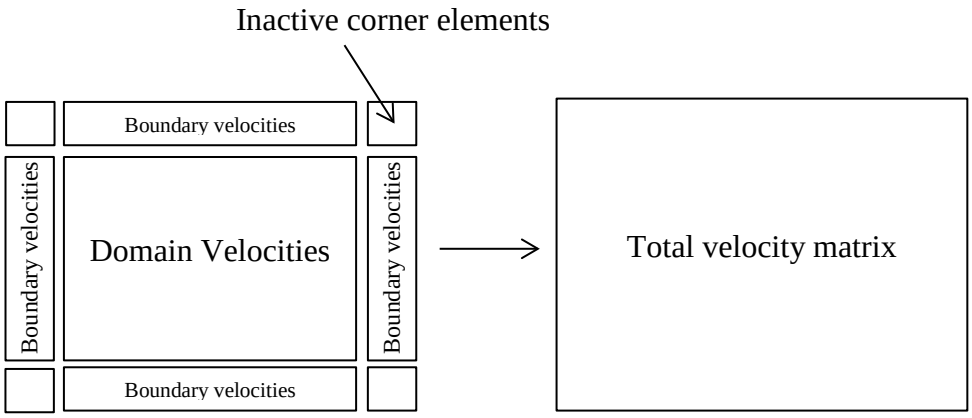


Figure 3.2: Assembly of domain velocities with boundary velocities

The first step in this calculation is to solve the momentum conservation equations in order to obtain the intermediate velocities. Equations (3.54) and (3.55) were solved iteratively. The equations were arranged to solve velocity implicitly so the iteration begins with the initial values of velocity, which is zero, for the first calculation cycle and in later cycles, the current available velocity was used to commence iteration. The newly calculated velocity vector was multiplied by an under relaxation factor (r) to update the velocity according to Equation (3.32). The use of a lower under relaxation factor confirms the convergence while extending the calculation time, but the use of larger under relaxation factors can lead to a divergent solution and cause errors. The optimum under relaxation factor depends on the nature of the solution; for rough rock fracture flows, moderate relaxation factors around 0.5 are suitable for general rough fractures but for higher roughness fractures with a variable aperture, those spans in a wide range should use lower relaxation factors. Best practice is to use the trial and error method to find the optimum relaxation factor for a particular rock fracture.

$$V_{update} = V_{previous} \times (1 - r) + V_{calculated} \times r \quad (3.1)$$

Once the intermediate velocity fields have been calculated the process proceeds towards correcting the pressure, so Equation (3.68) was solved iteratively to find these pressure corrections. The corrections for the inlet and outlet boundary pressures were set at zero to make the inlet and outlet pressure boundary conditions constant (fixed). According to Equation 4.1 the calculated pressure corrections were under relaxed and added to the pressure field to update the pressure.

At the same time Equations (3.64) and (3.65) were solved to find the velocity corrections and ensure continuity to the momentum conserved equations. Here the velocity corrections calculated were under relaxed by directly multiplying them with the under relaxation factor and then adding them to the velocity field to update. Up to this point one cycle of calculation has been completed, so the new velocity and pressure fields were returned to the beginning of the calculation to start a new cycle. This procedure can be stopped when all the corrections calculated (the pressure and two velocity components) become negligible compared to the value of the variable. A schematic presentation of this calculation procedure using the SIMPLE algorithm is shown in Figure 3.9, and the complete programme code is given in Appendix A.

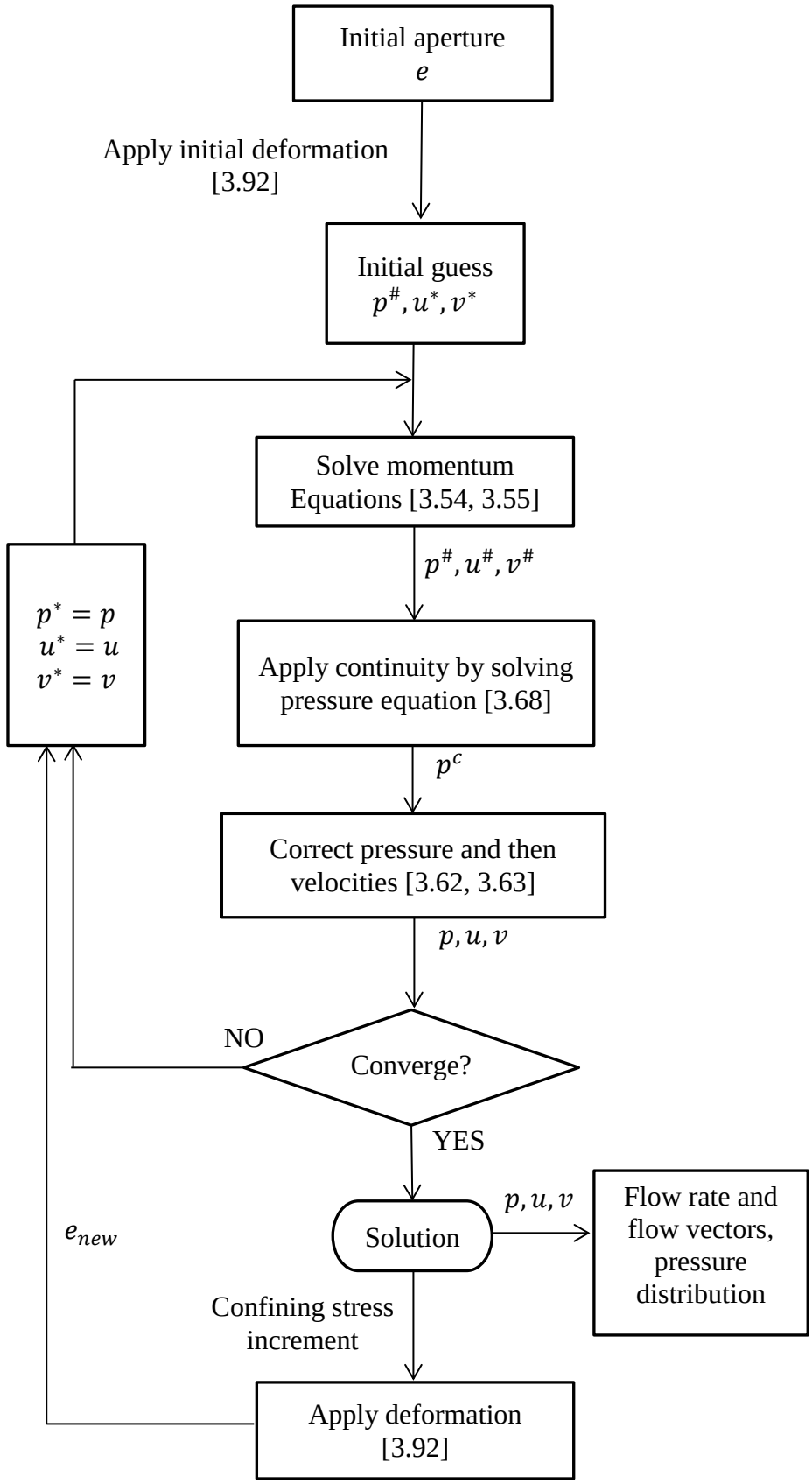


Figure 3.3: Flow chart for calculating pressure and velocity components

Once the velocity and pressure for the initial confining pressure has been solved, the solution can be presented graphically using a velocity vector plot. The velocities in 2D space are shown using arrows for the direction, while their relative magnitudes are shown by the length of the arrow. The background of the velocity plot is a filled contour plot of either aperture or pressure distribution (Figure 3.10 – pressure distribution, Figure 3.13 – aperture distribution). The user should choose this option before commencing the calculation.

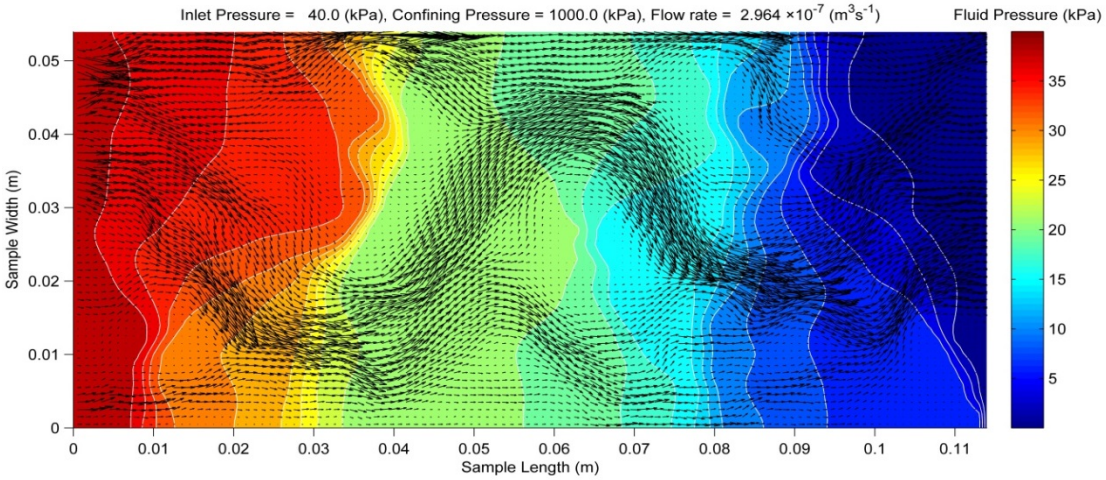


Figure 3.4: Typical solution output with velocity vectors and pressure contours at the background

Next, the increments of confining pressure are considered. If the user wants to calculate the flow behaviour when the confining pressure has increased, the user is asked to enter the increment interval of the confining pressure and the number of increments at the end of the first calculation for the initial confining stress. Then, according to the confining pressure increment, the deformation of the aperture can be calculated by using the current applied effective pressure (Equation 3.92), after which the deformations are added to the aperture matrix and the updated aperture is

used to calculate the new velocity and pressure distribution. The current solution for the velocity and pressure are used as the initial values for them, but since these values are already satisfy the continuity and momentum conservation for the previous aperture distribution, the next solution can be reached faster than the first one. At the end of each calculation cycle for a confining pressure increment, the velocity and pressure/ aperture distributions are presented graphically. At the end of the confining pressure increments, the output flow rate versus confining pressure relationship is presented. The user can save the solution data each time a graph or plot is presented for future analyses.

4.3.1 *Filled Contour Plot and Velocity Vector Plot*

The pressure and velocity output contains a contour plot in the background for the pressure distribution or aperture distribution. This plot was created using the function '*contourf( )*'. This function requires a matrix that contains the values of the pressure or aperture in 2D space. The positions in 2D space can be given by the same size two matrices that contain X coordinates in one matrix and Y coordinates in the other (the *'meshgrid( )'* function was used to generate these matrices). If the position matrices are not given, the values in the data matrix are plotted on an equally spaced 2D space. Additionally, the space between contours, including the colour and other line specifications, can be given as optional parameters, but if they are not given, the default values are used. In order to add a second plot to a current plot, the command '*hold*' should be used, and then a velocity vector plot is created on top of the filled contour map. The function for this is '*quiver( )*', and it plots the resultant velocity when the component velocities are given. The resultant velocity base is kept at the centre of the cell and the direction of the arrow is the resultant velocity direction. The

length of the arrow is scaled according to the range of velocities in the domain. The user can multiply the length of the arrow by a factor to increase its length or decrease its length as an optional parameter for the '*quiver( )*' function. The colour and line specification of the arrows can be defined by entering them as a parameter. Since the velocity components are stored on the faces of a cell, they should be averaged to the centre of the cell before being used in the *'quiver( )'* function.

4.4 Graphical User Interfaces (GUI)

This programme consists of graphical user interfaces for the input data as well as for output data, which makes the program user friendly. The main opening GUI of RFFS is for basic user inputs, preference selections, and execution of calculations (Figure 3.11). To begin the simulation the user must have an aperture matrix for the flow domain as a DAT (*.dat) file or Matlab data (*.mat) file. Apart from that the basic properties of the rock fracture and fluid should be given. The user can select the number of cells (finite volumes) in the X-direction and the RFFS calculates the number of cells needed in the Y-direction to preserve the aspect ratio of the input aperture matrix.

The boundary conditions and values should be entered in this GUI. The inlet and outlet boundaries are constant pressure boundaries, although the inlet pressure should be greater than the outlet pressure. Flow in the opposite direction can be simulated by setting the inlet pressure lower than the outlet pressure. The lateral boundaries should be made to 'wall' if membranes are assumed, as in the triaxial test. Having selected the lateral boundary as a 'mirror' where the fluid passes through the boundary by assuming that the velocities at the boundary are equal to the adjacent velocities.

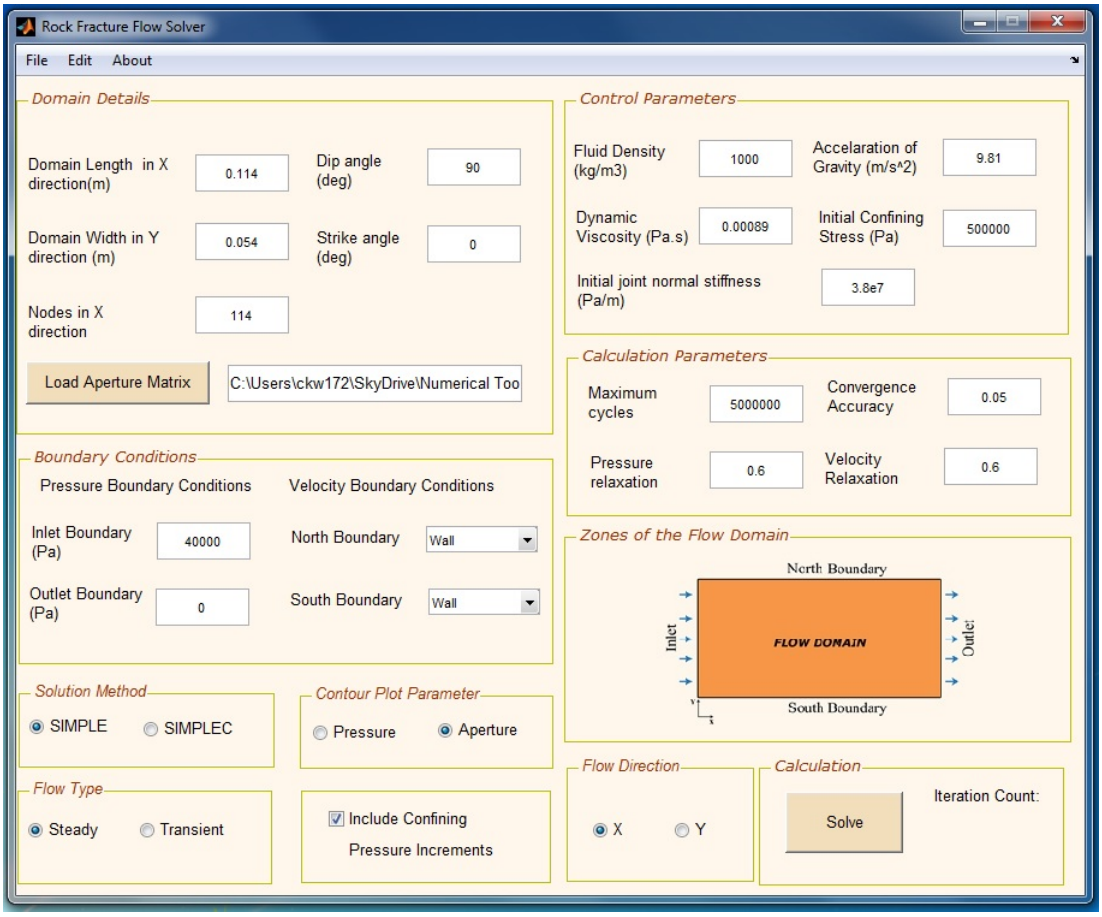


Figure 3.5: The opening graphical user interface

'Maximum cycles' in the calculation parameters is the number of iterations to be performed until convergence is reached. If the solution did not reach the convergence after that number of iterations, the user is asked to try additional iterations if needed or the calculation can be aborted. Maximum cycles are used in order to avoid infinite iterations due to unchanging solutions. Accuracy of convergence is the percentage error of the consecutive solutions in a row. If all the values in the velocity component matrices and pressure has an error percentage that is less than the given convergence accuracy, the calculation is stopped. Relaxation factors are used when variables are updated with the new value. Since the new value might not be the correct solution at the initial stages of iteration either, it is better to

have an appropriate relaxation factor to stop the solution from diverging. Once all the information has been entered the solve button can be clicked to execute the calculation. At the end of the calculation a velocity vector map is displayed (Figure 3.10, Figure 3.13).

If the user has selected to increase the confining pressure, after the velocity vector plot for the initial confining pressure, the incremental value of the confining pressure and the number of confining pressure increments can be entered in the pop up input boxes. Then the calculations are performed for the new confining stress and a new deformed aperture distribution is calculated. At the end of each confining pressure calculation, the current plot is replaced with the new velocity distribution. The plots can be saved at preferred confining pressure intervals if needed. At the end of the confining pressure increments, the flow rate versus confining pressure graph is plotted (Figure 3.12). Once the graph is plotted the related data can be saved to a file from the graph.

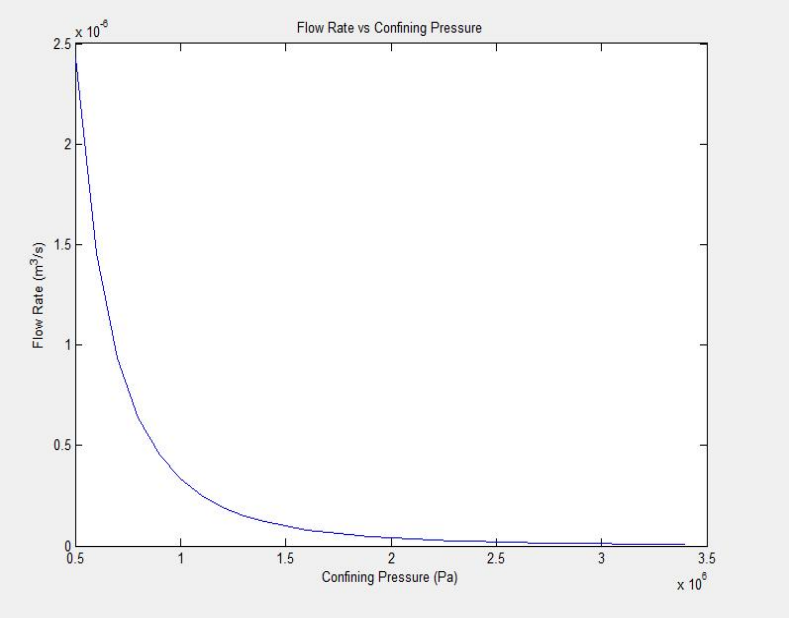


Figure 3.6: Typical flow rate vs confining pressure graph produced by the programme.

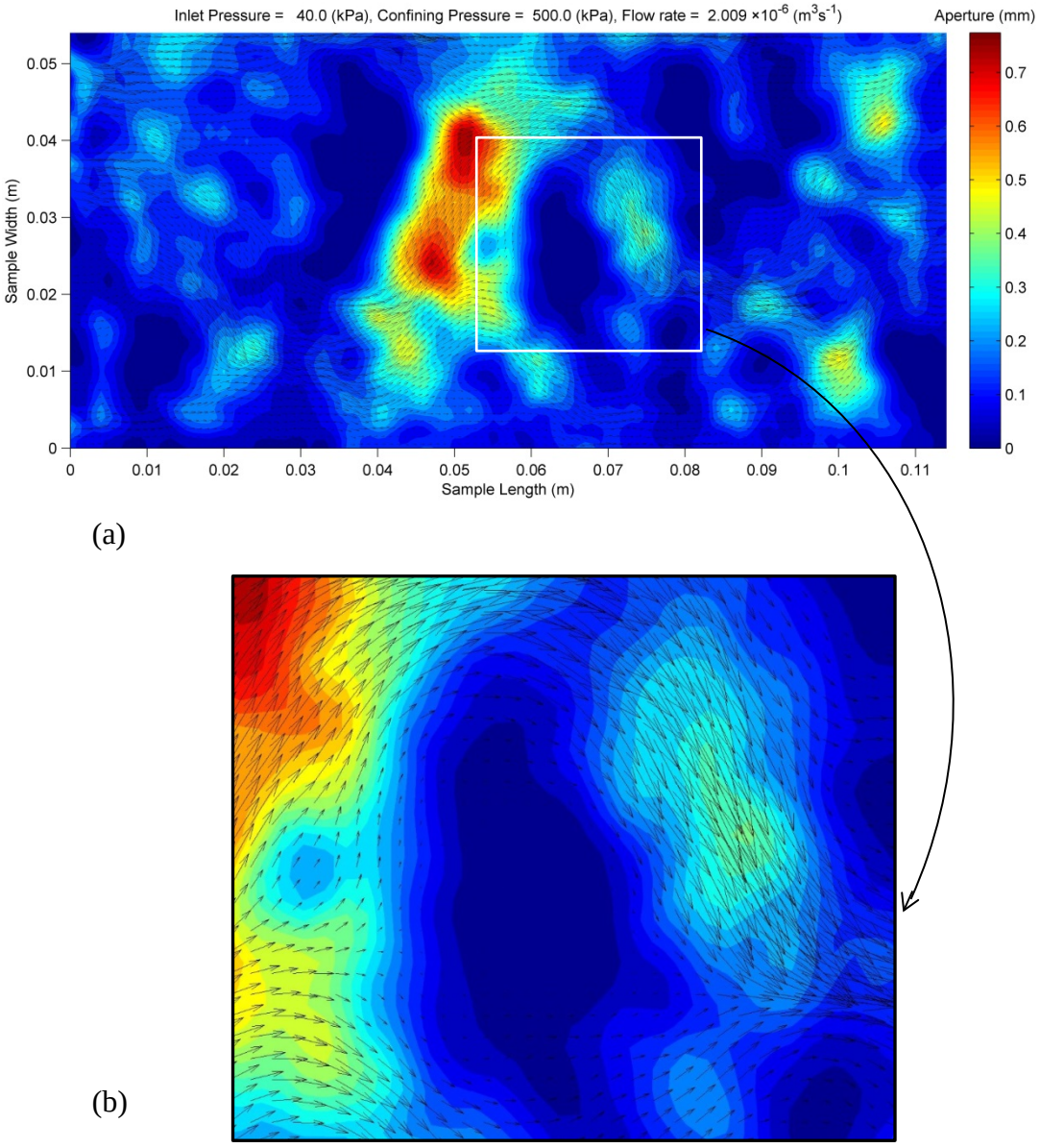


Figure 3.7: (a) A typical velocity plot with aperture distribution in the background

(b) Enlarged view of a part of (a)

4.5 Treatment of Contacts

If the cell *(I,J)* is a contact or the aperture of the cell *(I,J)* is zero, the four velocity components $U_{i,J}$, $U_{i-1,J}$, $V_{I,j}$, *and* $V_{I,j-I}$ will become zero because the faces of the cell *(I,J)* has become local boundaries (Figure 3.14). Furthermore, the pressure gradients

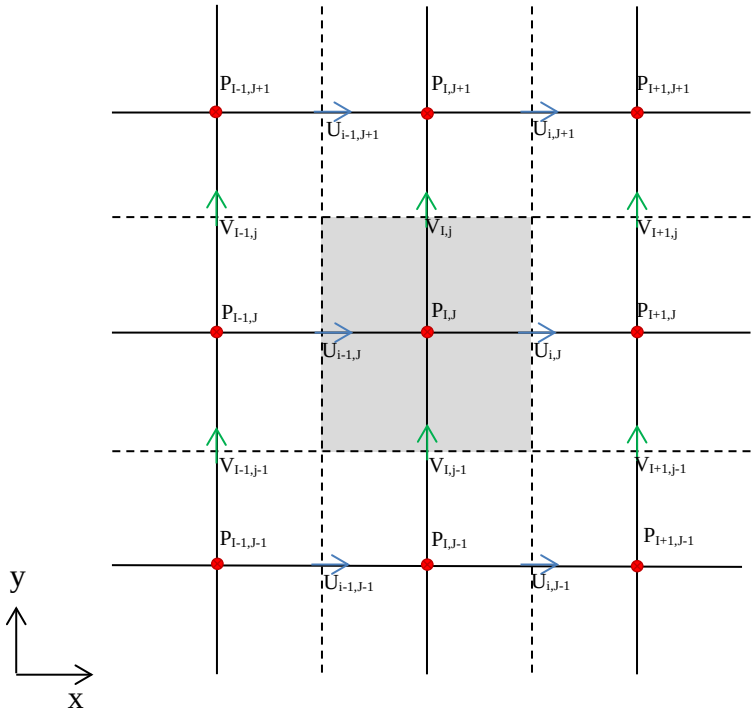


Figure 3.8: A contact cell *(I,J)* in the flow domain

perpendicular to these cell faces also becomes zero. When the momentum equations and pressure equation have been solved, they should be taken into account to treat the contacts. First, a list of the contacts in the domain should be given; this can be found by selecting the addresses of the cells where the aperture is zero, and then the corresponding affecting velocity components and pressure gradients can be found and modified for use in the calculation process. Figure 3.15 presents the modified SIMPLE algorithm to treat contacts.

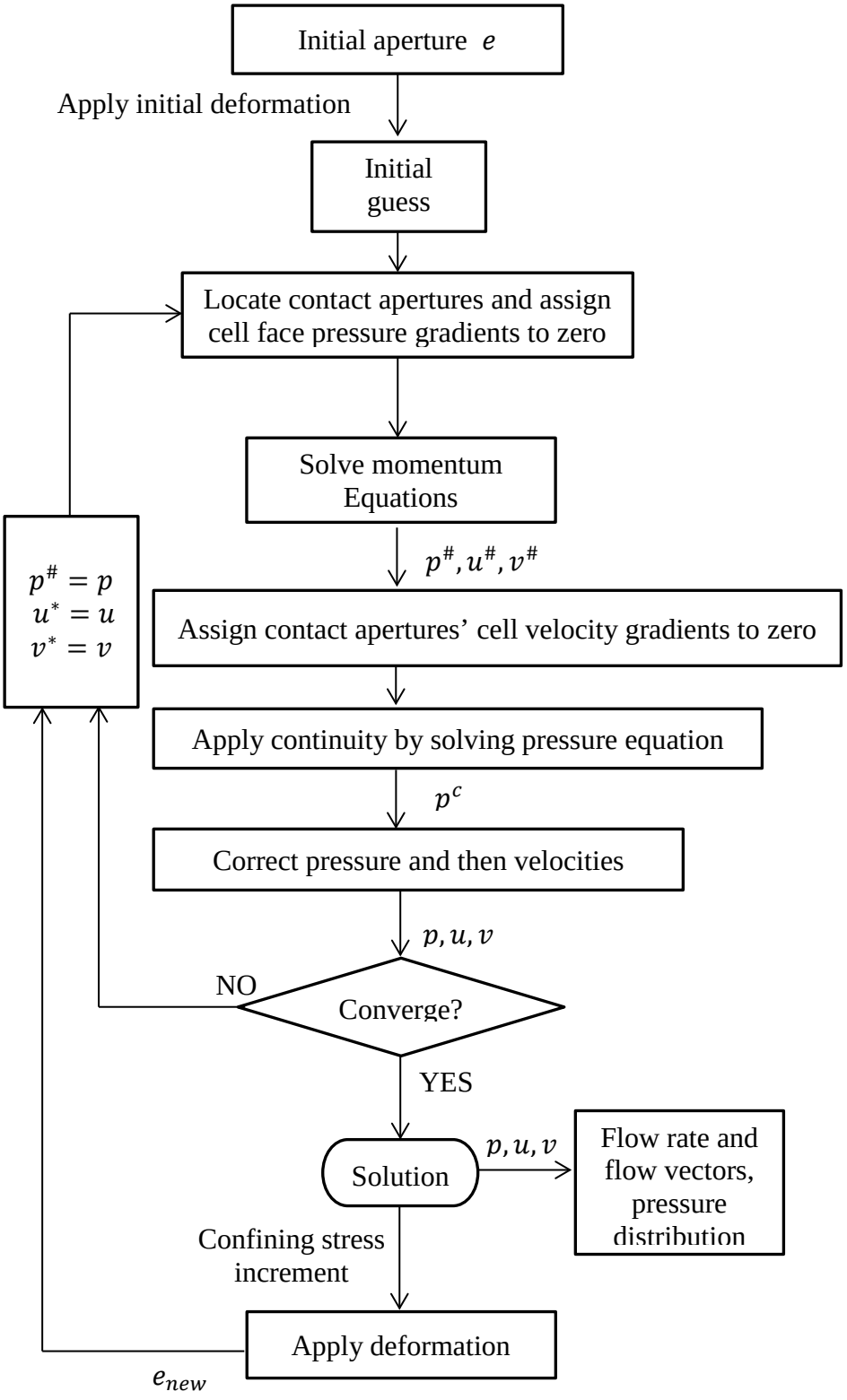


Figure 3.9: SIMPLE algorithm flowchart coupled with deformation calculation modified

A function was created named *localb(e)* (Appendix A) which receives the aperture matrix and returns the indices of velocity components which became local boundary values because of the zero apertures. This function is used when the momentum equations have been solved. Though the pressure of the cells was made zero for contacts, this will not help when calculating the pressure gradients at the contact faces, and therefore the cell pressure should be made equal to the neighbour pressure for the pressure gradient calculation. However there are four possible neighbours for a cell and therefore the cell pressure cannot be one value to make pressure gradients at the cell faces to be zero. When solving the pressure equation, four neighbouring matrices are formed according to the type of neighbour, i.e., east, west, north and south. In Figure 3.14, cell P(I,J) is the east neighbour for cell P(I-1,J) and the west neighbour for cell P(I+1,J), and so forth. The function *pneibr(PCB)* (Appendix A) will return the four neighbouring matrices for a given aperture matrix having zeroes for contacts. The returned matrices contain the pressure that the contacts cells need to make the pressure gradient zero at the contact cell face.

4.6 Calculation of Transverse and Opposite Direction Flow

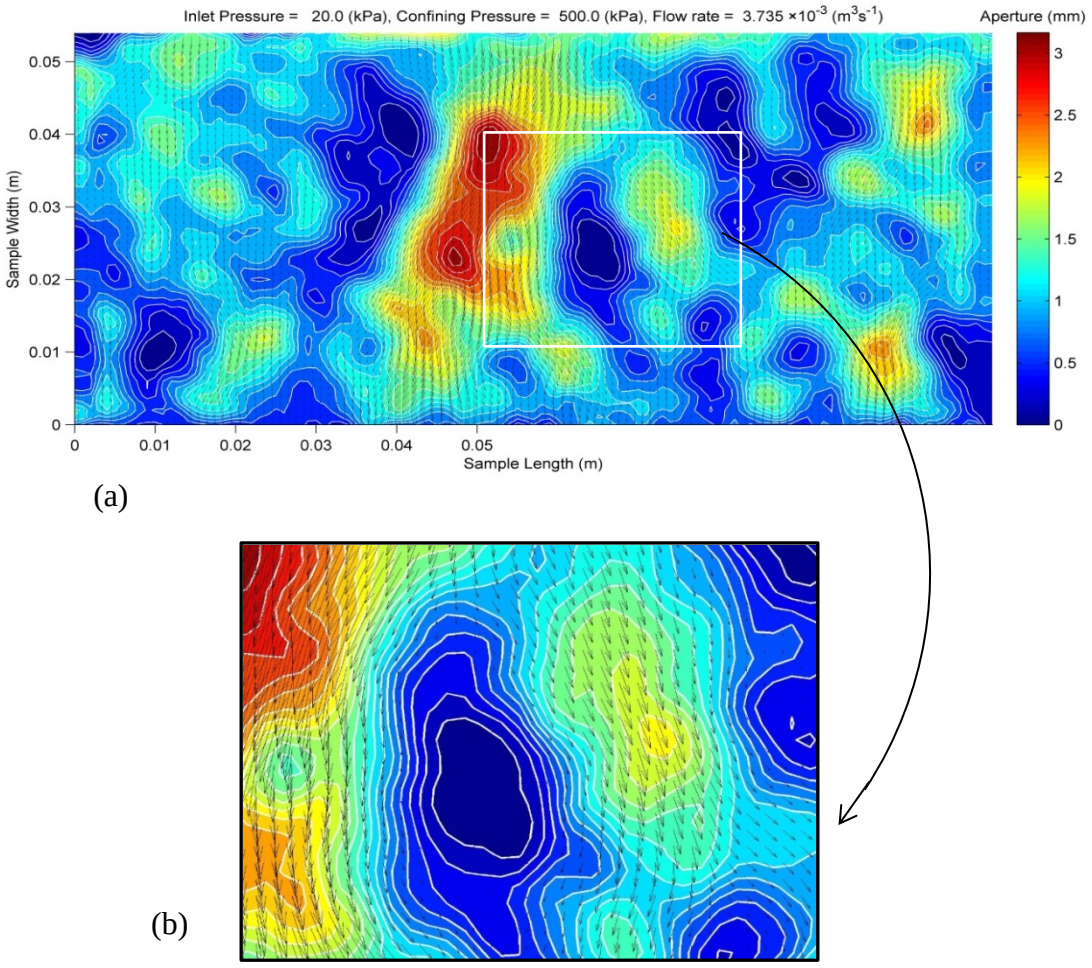


Figure 3.10: (a) Flow simulation in the transverse direction

(b) Enlarged view of a part of (a)

The usual triaxial cylindrical samples cannot be tested for transverse direction when determining the permeability matrix of a fractured network, it is essential to know the permeability of a fracture in either direction as well as the opposite directions for the determination of the permeability matrix. In this numerical simulation programme, it is easy to calculate those parameters by turning the aperture matrix

appropriately, and by interchanging the inlet and outlet pressure boundaries, flow in the opposite direction can be simulated. RFFS allows the user to select the usual direction (X-direction - Figure 3.13) or the transverse direction (Y-direction - Figure 3.16) as the flow direction and when selecting the boundary pressures, forward or backward (opposite) flow can be simulated. Turning the matrix and related display alterations can be found in the function for solving the SIMPLE algorithm (Appendix A).

4.7 Handling Errors

There are different types of errors in mathematical calculations. In a numerical solution scheme, there can be discretisation errors and round off errors. The differentiations of the governing equations are approximated using discretisation methods such as finite volume or finite different methods. These approximations always carry a truncation error which can be minimised using fine grid arrangements, but the magnitude of the grid spacing should be large enough to avoid any numerical errors in calculations. Round off errors are generated according to the selection of floating point numbers of a variable and the nature of the operation. Using variables with double precision floating point numbers can reduce round off errors but they occupy a larger computer memory. Furthermore, when arranging operations, it would be a good practice to predict the magnitude of the variables related to the operation. Subtracting numbers that are too close can produce zero errors while adding two numbers with larger difference can lead to smaller number being omitted according to the magnitude of the largest number. These errors can be minimised and eliminated by following proper programming practice.

There can be errors that are generated occasionally and whose occurrences cannot be predicted beforehand. For such unpredictable impossibilities, error catchers should be placed where they might occur. As an example, calculating the square root of a negative number and division by zero are two of those impossible operations. The usual practice is to check the particular variable before the calculation to determine whether it is negative or zero. If there are a lot of variables to be checked, this practice will cost more computation time in a program, but in MATLAB, these errors can be found once the operation has taken place. MATLAB returns impossible calculations as 'Not a Number' (NaN) and infinity as 'Inf' or '-Inf' as the result, so by checking the result for 'NaN' or "Inf', the proper remedial measures can be determined. 'NaN' and 'Inf' are treated as constants and can be referred to and searched for in the variables. There are other inbuilt functions in MATLAB to check usual errors that can occur in a calculation; the following error checks are the main ones used in this calculation of RFFS.

4.7.1 *Existence of the Aperture Matrix Data File*

The function *"exist(path, 'file') "* checks the existence of a file located in the directory given by the string variable 'path'. When the user enters a non-existing file this function finds it and returns '0' or 'false', which can be checked, and displays the error aborting the calculation.

4.7.2 *Divergence Check*

When the relaxation factors are not small enough or the given parameters do not have an existing solution, the velocities and pressure calculations can begin to diverge producing 'NaN' (not a number) as the result. It is better to stop the

calculation immediately if a result contains an 'NaN' value because they multiply in the next iteration and further calculations have no use. The function '*isnan( )*' was used to check whether a matrix contains at least one 'NaN' combining with the function '*any( )*'. To check whether the U velocity component will diverge, '*any(isnan(U))*' was used in this programme, and if this returns 1, that means there is at least one 'NaN' value in the matrix U and then the calculations are aborted.

4.7.3 *Convergence Check*

Convergence is assumed to be reached when all the elements of the correction matrices are negligible. This can be checked by using the function '*all( )*'. All the elements of the matrix can be checked to determine whether they are less than the threshold number. This function will return 1 if all the elements satisfy this condition.

4.7.4 *Emptiness Check*

The function '*isempty( )*' is used to check the user inputs before processing. If users fail to enter any data and begin the calculation, the emptiness of the input value is checked before starting the next calculation.

4.8 Data Access and Storage

RFFS receives aperture data as a DAT file or MAT file. Creating a DAT file for the aperture data can be done using a text editor. Alternatively, aperture data can be saved as MAT files in MATLAB and RFFS can read them. Currently, RFFS is not formulated to receive any other data formats. Output data can be saved by the user if needed, and they will also be saved into a DAT file. The velocity plot figures and other graphs are saved in JPEG format. A figure is set to a resolution of 1200 dpi.

Alternatively figures can be saved as MATLAB figure (*.fig) files or enhanced meta files (*.emf). Matlab figure files can be opened in MATLAB again for any further analysis and enhanced meta files are suitable to resize when use in text editors.

4.9 Summary

This chapter has described the development of a new computer programme, RFFS: Rock Fracture Flow Solver, using MATLAB programming language to execute the numerical solution procedure described in Chapter 3. RFFS consists of graphical user interfaces to interact with users to receive input data and deliver the output. Velocity components, and the aperture and pressure are stored in matrices, and the calculations are performed iteratively until a stable solution has been found. Special treatment for contacts has been done and extensions have been made so that flows can be calculated in reverse and transverse directions.

Chapter 5 Laboratory Experiments

5.1 Introduction

Laboratory experiments are important in validating a new mathematical model, because, in a laboratory there is enough control on the model parameters to ensure accurate verification. In this study, a mathematical model was developed to calculate flow through a real rough rock fracture qualitatively and quantitatively which was presented in Chapter 3. This chapter outlines the laboratory apparatus and test procedures used to validate the proposed mathematical model.

The triaxial apparatus is used on samples of soil and rock to test their strength and permeability. The salient feature of the triaxial apparatus is its ability to apply stress on all three principle axes. Of the triaxial apparatuses available, high pressure triaxial apparatus is used for rock samples. The High Pressure Two-Phase Triaxial Apparatus (HPTPTA) designed and built at the University of Wollongong was used to test rock fracture permeability in this study. The first part of this chapter presents the basic features and functions of HPTPTA and then describes how the experimental investigations were conducted using this apparatus. The second part of the chapter explains how the rock fracture apertures were measured. In this study a

non-contact laser scanner was used to scan the rough surfaces and obtain the aperture distribution of a rock fracture.

5.2 High Pressure Two-Phase Triaxial Apparatus (HPTPTA)

HPTPTA received its name because the apparatus can cope with two separate fluid inputs. This is a salient feature of this apparatus because two inputs can have different input pressures whereas a two-phase input mixture can only have one pressure (Indraratna and Haque 1999; Indraratna and Ranjith 2001; Zhang 2013). The main components of the HPTPTA cell are shown in Figure 5.1 (not to scale).

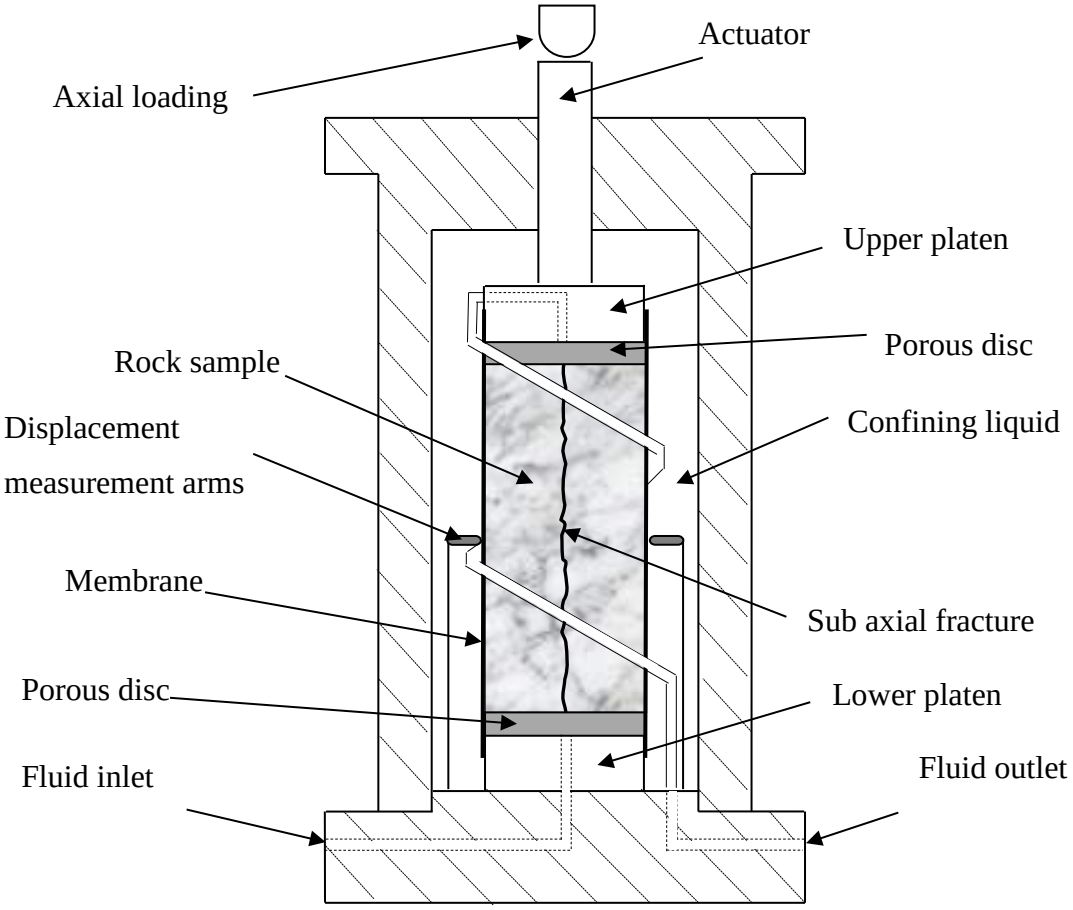


Figure 5.1: Components of HPTPTA cell with a rock sample

The triaxial chamber was manufactured from high-strength stainless steel to withstand high pressures; HPTPTA can be used to apply confining pressures up to 150 MPa. The triaxial cell has an inner diameter of 130 mm and the lower platen can be changed according to the sample size. Two platens were used in this triaxial cell to test 54 mm and 61.5 mm diameter rock samples because they are common core sizes in rock sampling. The maximum height of a rock sample that can be tested in the cell is 130 mm, but in triaxial testing the ratio of the height to diameter of a rock sample is around 1:2. A porous disc was placed between the sample and the platen to ensure equal flow to the sample and prevent the flow path clogging. As well as the main triaxial cell unit, the other support units are as follows;

I. Loading unit
II. Fluid inlet unit
III. Fluid outlet unit
IV. Data acquisition and recording unit

The arrangement of these units and their components are shown in Figure 5.2 and Figure 5.3 and show the experimental set up of the triaxial apparatus in the laboratory.

5.2.1 *Loading Unit*

The loading unit has two main components: axial loading and lateral confinement. Axial loading may be either stress-controlled loading or velocity-controlled loading. Velocity-controlled loading is used for the triaxial shear tests. In this study stress-controlled loading was used to apply the deviator stress. Lateral confining pressure was applied via silicon oil pressurised by a servo controlled hydraulic pump.

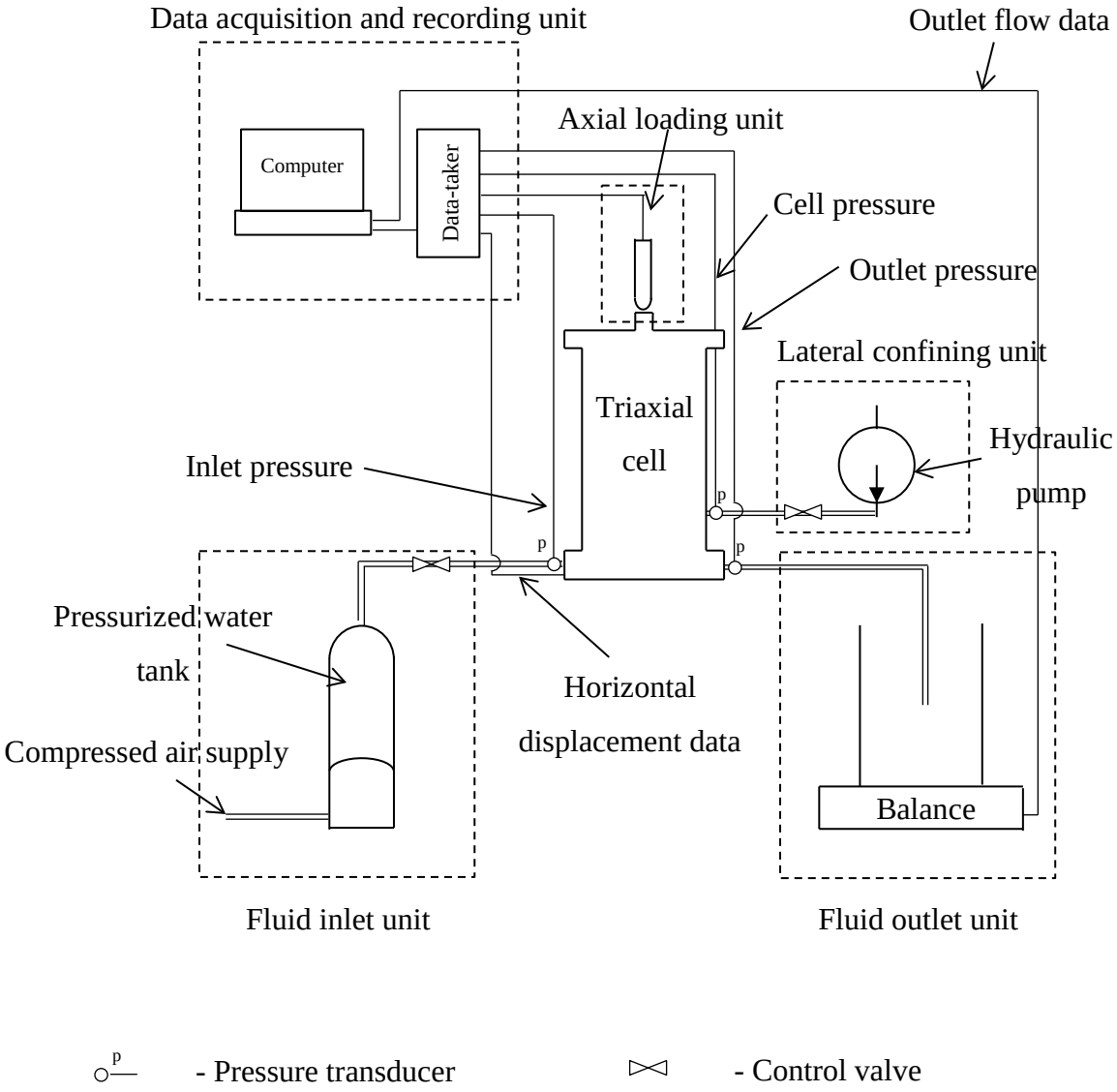


Figure 5.2: Units of high pressure triaxial apparatus

5.2.2 *Fluid Inlet Unit*

The fluid inlet unit consists of a water tank, compressed air supply, and a controlling valve. The water tank has two compartments separated by a diaphragm, one for water and the other for compressed air. Compressed air pressurises the water in the tank and the controlling valve adjusts the inlet water pressure by adjusting the inlet air

pressure. A pressure transducer is connected at the inlet of the triaxial cell to measure the inlet pressure.

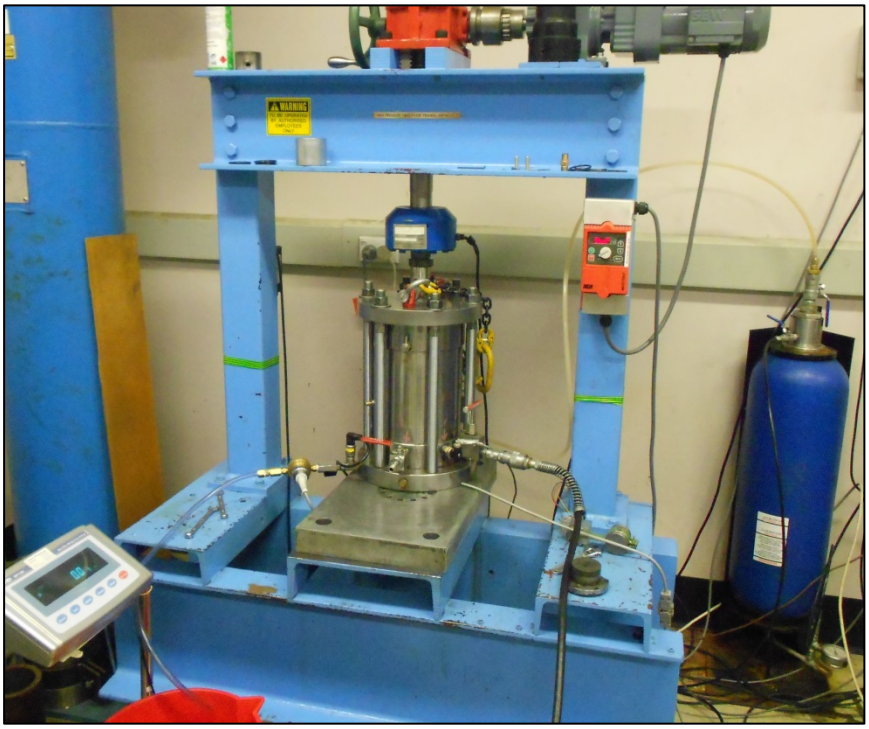


Figure 5.3: Triaxial apparatus (HPTPTA) set up for a test

5.2.3 *Fluid Outlet Unit*

Outflow of the triaxial cell is directed to outlet flow rate measurement unit. The outlet flow is collected in a flask mounted on an electronic balance, and the weight of the flask containing outlet water is recorded at prescribed time intervals to calculate the flow rates. The outlet pressure is recorded with a pressure transducer.

5.2.4 *Data Acquisition and Recording Unit*

This unit consists mainly of a data taker (data logger) and a computer. Pressure transducers and a horizontal displacement measuring device send continuous signals to the data taker. The data taker was programmed to convert the voltage signals into the required units after the sensors had been calibrated. Data was recorded at

prescribed time intervals in a memory card of the data taker and transferred to the computer at the end of the test. The weight of the outlet flow was recorded at a same time as the data taker and transferred to the computer at the end of the test.

5.2.5 *Calibration of Sensors*

The sensors had to be calibrated to ensure correct data readings, so the pressure transducers and the horizontal displacement measuring device were calibrated as follows.

5.2.5.1 Pressure transducers

The pressure transducers were calibrated with a dead weight tester (Figure 5.4). Known pressures were applied with weights and the voltage response of the transducer was recorded. The linear relationship that was developed between the voltage response and the pressure was used to convert the sensor data into pressure in kPa.

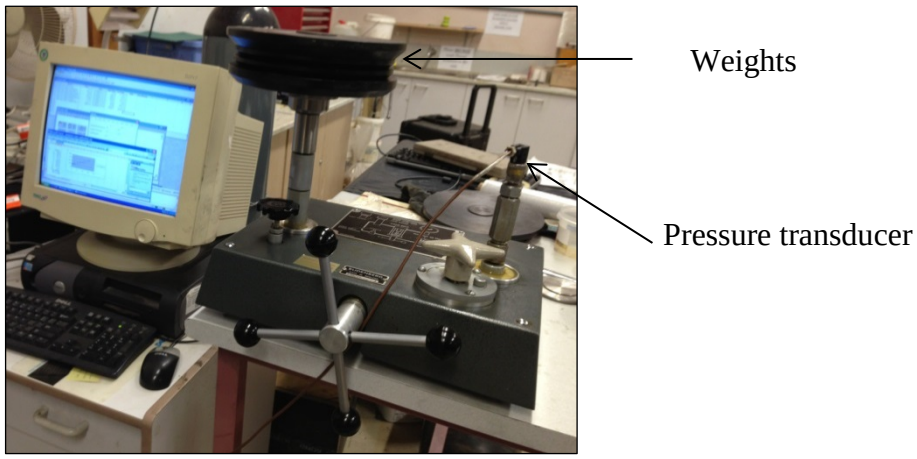


Figure 5.4: Dead weight tester

5.2.5.2 Horizontal displacement measuring device

Horizontal displacement was measured using two cantilever arms. The vertical part of the arm is a flexible metal plate with four strain gauges that are connected to each other to form a Wheatstone bridge; four gauges were used to ensure that the deformation data was reliable. A micrometre screw gauge was used to calibrate the cantilever arms.

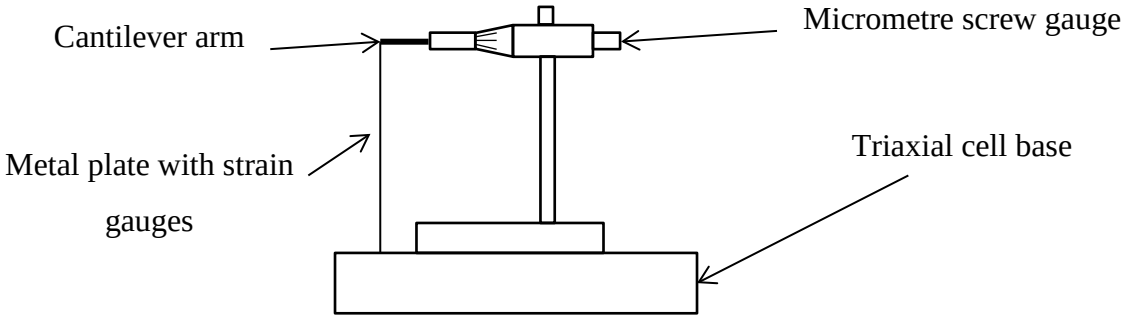


Figure 5.5: Calibration of horizontal displacement measuring device

The cantilever arm was kept in the usual place in the triaxial cell and the micrometre screw gauge was fixed so that it faced the cantilever arm (Figure 5.5). The frame of the micrometre screw gauge was not used in this process. Small displacements were made by the screw gauge ratchet and the corresponding voltage responses from the strain gauges as the metal plate bent were recorded. The datataker was then calibrated to convert the signal into millimetres by assuming a linear relationship between voltage response and displacement.

The displacement captured by the cantilever arms contained the deformation of the membranes; an error that should be removed from the data received to calculate displacement of the rock fracture. Zhang (2013) conducted deformation tests for the membranes made at the University of Wollongong (describe in the

section 5.3) and developed a relationship between membrane deformation and applied confining pressure. These tests were conducted by replacing the rock fracture with a rigid steel cylinder. Membrane error was corrected by equation (5.1) that was developed from these experiments.

$$\delta_m = 0.1963 \ln(\sigma_n) - 0.8511 \quad (5.1)$$

Where δ_m is the total deformation of the membrane on both sides, and it is perpendicular to the axis of the sample in mm, and σ_n is the applied confining stress in kPa.

5.3 Sample Preparation

5.3.1 *Creating a Fractured Rock Sample*

Cylindrical rock samples were cored out from sandstone blocks using the drill rig in the laboratory. The sample was 54 mm in diameter and more than 110 mm long, but it was later trimmed to the required length with a diamond saw. The rock samples

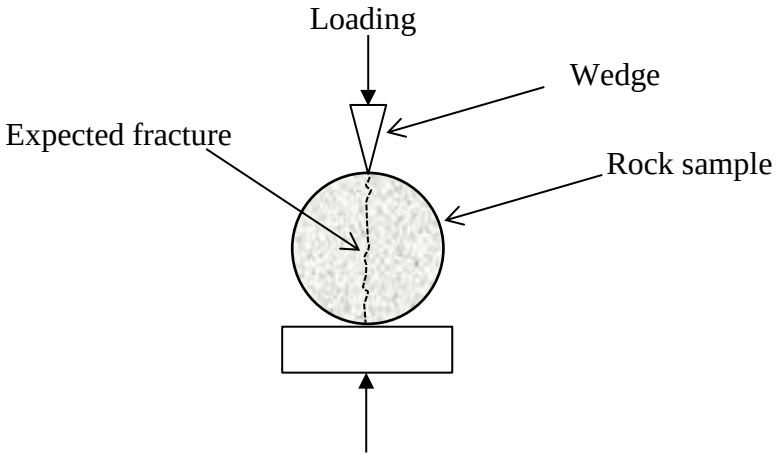


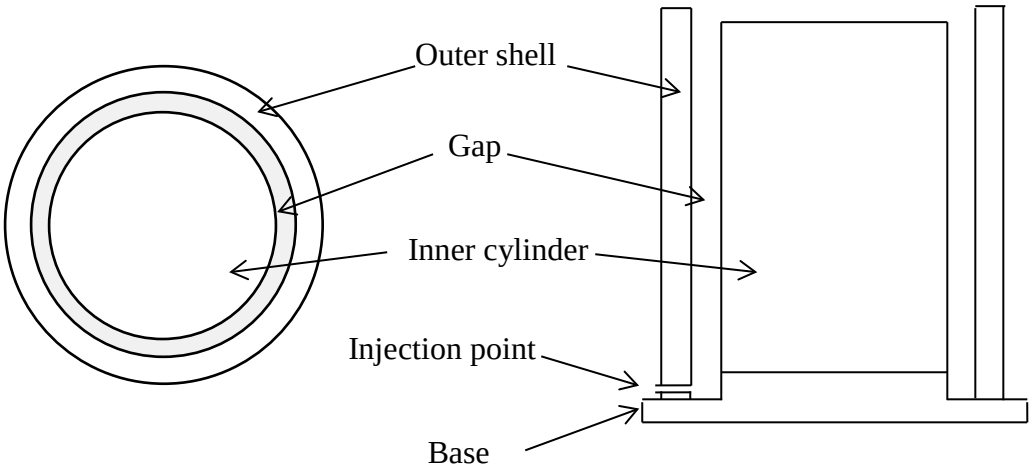
Figure 5.6: Wedge splitting arrangement

were then split with a sharp wedge arrangement (Figure 5.6) to ensure a sub-axial fracture. To obtain a mismatched surfaced fracture, each half was trimmed separately and equally at opposite sides.

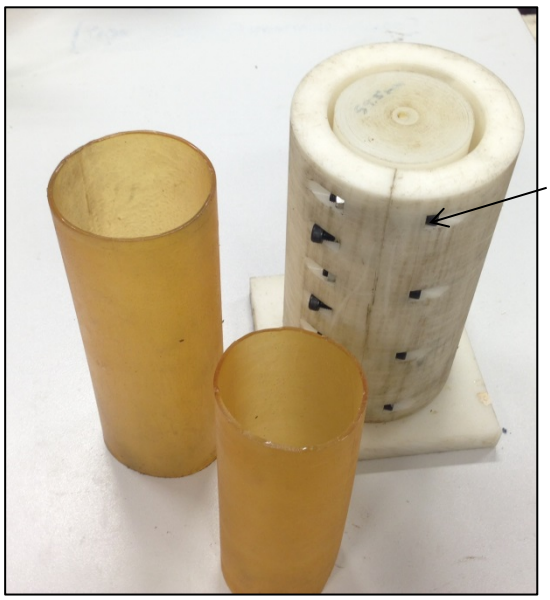
5.3.2 *Preparation of Membranes*

The membrane separates the confining fluid from the sample while the pressure from the confining liquid was transferred to the sample. In permeability tests the membrane prevents the permeating fluid (water in this test) from mixing with the confining fluid. The membrane for the test should be in good condition because high pressures are applied. Most thin membranes tend to puncture at high confining pressures because the rock sample had coarse edges and surfaces. Therefore, 2 mm thick membranes were made in the laboratory. A special mould and a fast setting mixture of chemicals were used to create the membranes. A commercial polyurethane solution called F50 (Barns, Australia) which contains two parts, was used to cast the membranes. Parts A and B should be mixed in a proportion of 1:2 by weight and injected into the mould within half an hour of being mixed. Air bubbles can be trapped in the mixture because its viscosity is higher, so after the stirring the mixture was placed in a vacuum chamber to remove any air bubbles. If an air bubble remains in the membrane, it will create a hole, which is why de-airing is an essential part of this casting procedure. Air bubbles can become trapped while stirring the two parts together, so do not lift the stirrer up while stirring and use a moderately constant stirring speed to reduce the number of trapped air bubbles. When the mixture is being placed inside the injecting syringe, bubbles of air may become trapped in the mixture. To avoid this scenario, the syringe needle should always be submerged in the mixture and topped up before the mixture in the mixer runs out.

Finally, the mixture should be injected into the mould at a slow but constant speed and close the injection point at the end.



(c) (b)



Screw open arrangement to unmould the membrane

(a)

Figure 5.7: Components of membrane mould (not to the scale). (a) Plan view (b) Side view (section) (c) Prepared membranes and the mould

The mould contains three components, the base, the outer shell, and the inner cylinder. The inner cylinder is slightly smaller in diameter (53.5 mm) than the rock

sample and the inside diameter of the outer shell was 55.5 mm in order to have a 2 mm thick membrane. The base holds the membrane mould components vertically while maintaining a 2 mm gap around them. The outer shell can be opened from one side to de-mould the membrane once it has been set. The membrane can be de-moulded after 24 hours but it should then be allowed to rest for 7 days to attain its maximum strength.

5.4 Rock Fracture Surface Scanning

Rock fracture surface profiles are needed when modelling rock fracture flows in two-dimensions. The fractured surfaces were scanned using a non-contact 3D laser scanner (Minolta VIVID 910, Figure 5.8). This scanner has better capabilities for scanning and also comes with editing software which helps in preparing the final 3D scan. The main advantage of this scanner is that no positioning table is required for scanning. Ordinarily a 3D scan cannot be completed in one scan because the laser beam can only see part of the 3D object when scanning, and therefore at least 3 scans are needed to complete a scan. However, when an object is not symmetrical and smooth, more than 3 scans are needed to complete a 3D image. To synchronise each scan means that the positioning tables are usually needed, but this 3D scanner can merge two scans together by pointing at approximately 3 unique positions in two scans. Each scan in this scanner can be merged to the current combined scan and complete the final 3D object, while unwanted areas can be removed and any noise or holes can easily be edited with the editing software (Geomagic, Qualify 12.1.2).

Figure 5.8: The 3D laser scanner

The scanning procedure is straightforward. There should be enough features in the scanning object to define the origin in order to obtain a surface profile. The easiest way of defining an origin is to have three orthogonal planes. Other combinations can be three points with known co-ordinates, or three orthogonal lines, etc. If the object does not have them, it should be mounted onto another object (a frame) that has three orthogonal planes or any other combination. This final arrangement can be turned to any position to scan. Since the semi-cylindrical shape of half the rock sample has not enough features to define an origin, the samples were kept on a reference block to scan.

The scanner was connected to the computer and the data received from the scanner was processed by the Polygon Editing Tool software. The scanning process can be done by the scanner controllers or remotely from the computer via the Polygon Editing Tool. Remote operation is easier with multiple scans for a single

object. When the first scan has been stored the next scan is stored after pointing three identical points in the previous scan and current scan. Once the new scan is stored, it will be directly aligned to the previous scan, which means the new scan and the old scan should have some common area in which to nominate three points. Therefore, the best practice for scanning is to always turn the object in one direction (clockwise or anti-clockwise) while allowing for some overlapping area to be scanned in the adjacent two scans.

5.5 Experimental Procedure

Once the rock sample was prepared the surfaces were scanned to obtain an initial surface profile and the sample was then then loaded into the triaxial cell. Before loading the sample, it was surrounded with the membrane such that an equal length of the membrane was on both sides. Then the lower platen of the cell was topped with a porous stone, onto which the sample was placed to ensure the membrane covered the lower platen. The sample was placed to align the fracture and ensure that the cantilever arms were perpendicular to the fracture plane, and then the lower platen and membrane jacket were tightened with a hose clip. The upper platen and porous disc were then placed on top of the sample, but not before inserting the hose clip for the upper platen in case the hose clip cannot open. After placing the upper platen, the membrane jacket was tightened with a hose clip and then the outlet tube was connected to the upper platen. This completed the sample loading procedure. The cell was placed to enclose the sample and was then tightened to the base of the cell. After ensuring that the base of the cell and cell fitted well, the confining liquid (silicon oil) was poured into the cell until it filled up. An actuator was then inserted into the cell from the cell top and the cell bleeding valve was kept open to allow

trapped air to escape from the confining liquid. Once the actuator reached the upper platen, the bleeding valve was closed. The whole triaxial cell was then moved towards the vertical loading unit until the actuator and loading piston were aligned, and then a vertical load was applied so that the rock sample could not move in vertical direction.

The next phase of the testing procedure was to make the fluid and data connections. The inlet and outlet valves were connected to the respective tubes and then the hydraulic pump was connected to the cell. All the pressure transducers and horizontal deformation measuring system were connected to the datataker and data logging was started. After data logging had started, the confining pressure was applied progressively to allow the initial deformations to be recorded as the pressure increased. Once the starting confining pressure had been reached, the inlet pressure of water was adjusted to the required amount and allowed to reach a steady flow. Once the flow became steady, the outlet weight was recorded at known time intervals until sufficient data was received. The necessary alterations of inlet pressure and confining pressure were then applied and all the data were recorded.

Once a sample had been tested, the fracture surfaces were scanned to check for any damage, and then the sample aperture was measured as described in the next section.

5.6 Initial Aperture Measurement

In these tests the initial aperture is the aperture distribution when no confining pressure has been applied. That is when the applied normal stress to the fracture is zero. This means the aperture should be measured without any pressure being applied perpendicular to the fracture. This measurement could be done by inserting an epoxy

mixture into the aperture and casting the gap between the two fracture surfaces (Indraratna *et al.* 2002). The weight of the upper half of the sample was omitted here because these samples were quite small so it was assumed that any pressure applied

Figure 5.9: Scanning a rock fracture surface

by the upper half of the sample could not deform the fracture.

Aperture distribution was measured after having replicated the aperture with an epoxy that was made by adding two chemicals together and leaving that mixture to settle. These chemicals are used commercially to replicate the fine details of objects and are therefore suitable for replicating an aperture distribution. This epoxy resin is usually invisible once set, but the scanner needs an opaque surface that a laser beam cannot penetrate to obtain the correct surface profile. This is why the resin is coloured before being used, but these coloured pigments should not react with the resin in order to cure the final epoxy. Silicone pigments were used to colour the resin because they are inert and do not affect the final product. Yellow was selected for these tests but the scanner still returned good results even when the surfaces were red or green. The two parts of the resin were mixed in a ratio of 1:1 in weight and once mixed a drop of colour pigment was added, but if that was not enough to make the mixture opaque, additional pigments were added. Before the resin was mixed, one half of the rock sample was kept on a stand with the fracture plane facing upwards and horizontal. After the resin was mixed it was poured onto the surface of the rock sample, and because its viscosity was high, the mixture remained on the surface without running over the sides. The upper half of the sample was then placed on the lower half, and the edges were aligned, but without any additional stress. This returned the rock fracture to its initial position and then the whole system was allowed to set for 24 hours. After that the upper half was removed and the lower half with the aperture replicated above was used to measure the aperture distribution by laser scanning.

5.6.1 *Determination of Aperture Distribution*

This procedure begins by selecting a reference frame. The basic idea of measuring the aperture is to scan the resin surface first and then scan the fractured surface in order to obtain the difference between the two surface profiles (Figure 5.9). To achieve this, both scans should be brought to a common origin. The sample did not

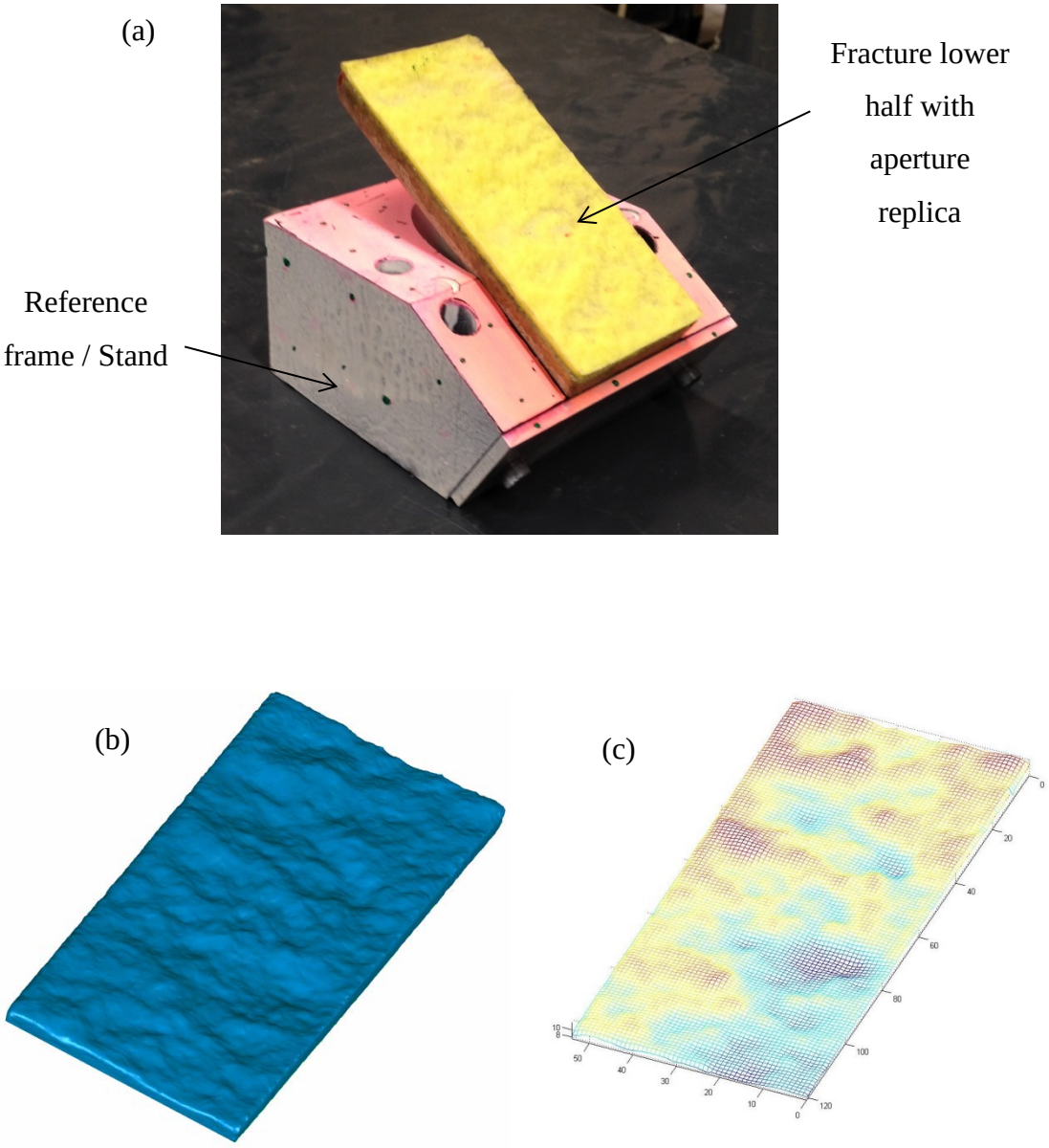


Figure 5.10: (a) Specimen kept on the reference frame for the scanning (b) Scanned surface (c) Reproduced wire mesh of a fracture surface

have enough details to define an origin for the surface profile because of its semi-cylindrical shape, so it was placed on a reference object which has three orthogonal planes in order to define an origin for the profile data. Figure 5.10 shows the sample kept on the reference frame or the stand, including the scanned fracture surface with its wire mesh representation, using the scan data.

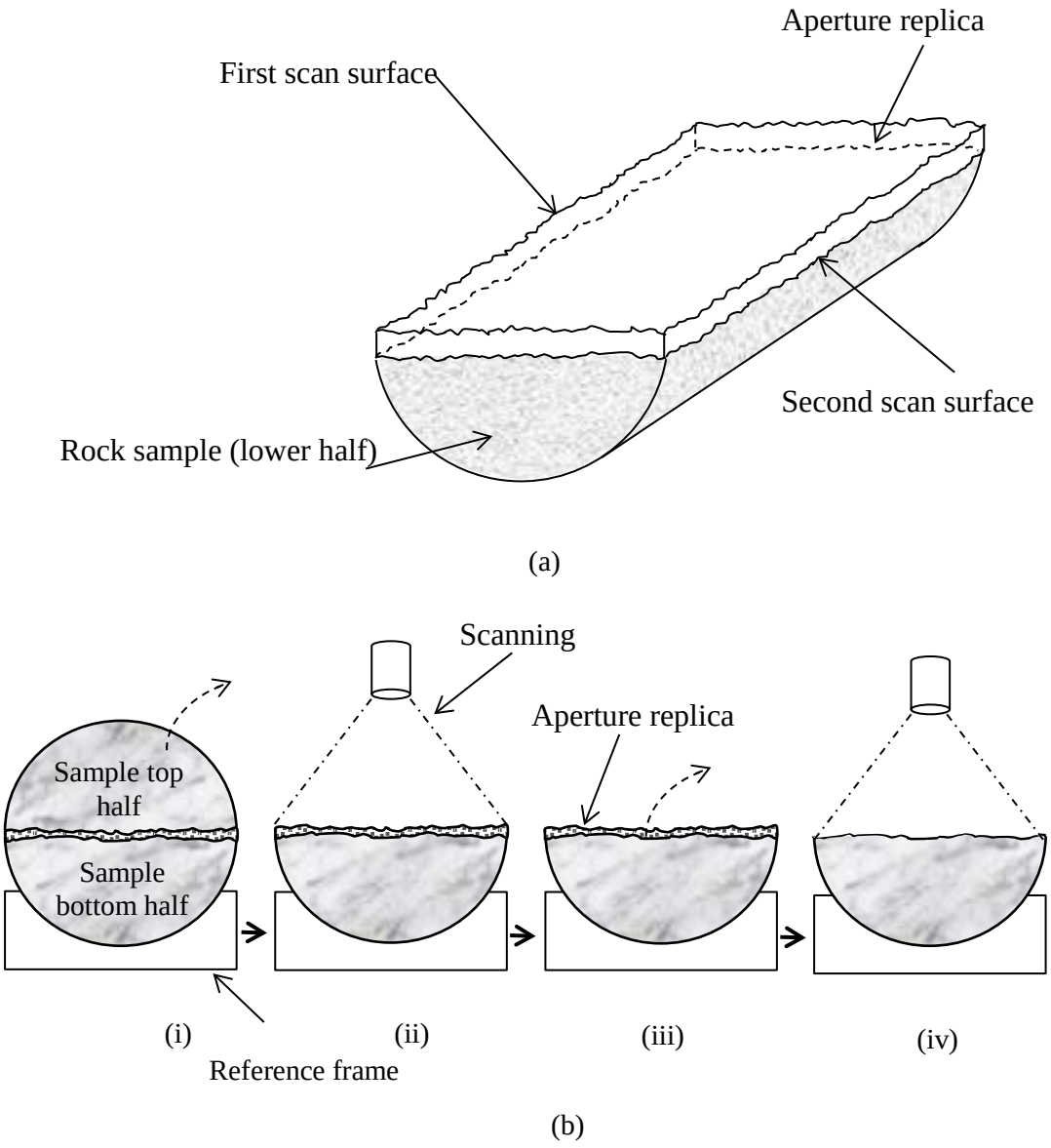


Figure 5.11: Aperture measurement (a) Components of scanning sample (b) steps in the scanning procedure

The aperture was measured by scanning the replicated surface of the aperture followed by scanning the lower half of the surface aperture. While conducting these measurements the sample should be fixed to the stand to ensure that the measurements are accurate. The epoxy replica was also carefully removed from the rock sample (Figure 5.11) to scan the fractured surface. Once the two surfaces were scanned, the editing software was used to obtain the aperture distribution.

5.7 Estimation of Aperture from Surface Profiles

The procedure for calculating the aperture, as explained above, takes time and a great deal of work in the laboratory, however, the aperture distribution is directly related to the two surface profiles of the rock fracture. At the initial and zero stress states, the two fractured walls come close together so that some taller asperities touch each other and prevent the joint from closing any further. Here, we can assume that having divided the fractured surface into four equal quarters, the highest asperities from each

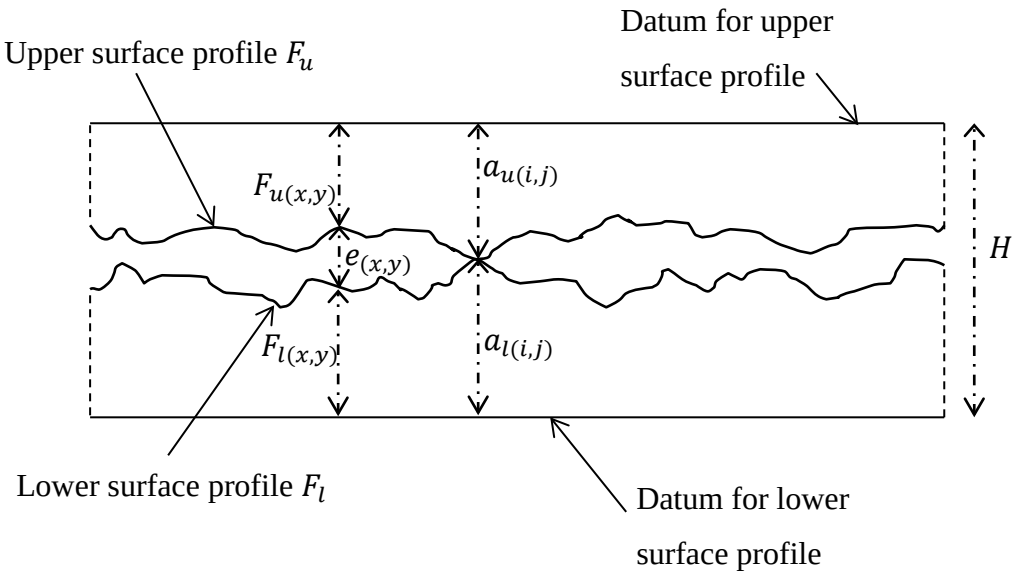


Figure 5.12: Aperture distribution between two surface profiles

quarter would touch and stop them closing any further when the two surfaces are brought together. In this manner, the position of the two surface profiles can be determined and then the aperture between the two surfaces can be calculated. Figure 5.12 shows the key parameters needed to calculate the aperture distribution from two surface profiles.

If $a_{u(i,j)}$ and $a_{l(i,j)}$ are the two prominent asperities that would meet first in a particular quarter (j) when the surfaces are brought together, assuming,

$$H_{Qj} = a_{u(i,j)} + a_{l(i,j)} \tag{5.2}$$

Then for all four quarters, a H_{Qj} value can be calculated and the average of the four is taken as the mean distance between the datum of the two surface profiles (H). An aperture ($e_{(x,y)}$) at any point of the fracture can then be calculatedby:

$$e_{(x,y)} = H - F_{u(x,y)} + F_{l(x,y)} \tag{5.3}$$

$F_{u(x,y)}$, and $F_{l(x,y)}$ are the surface profiles of upper and the lower fractured walls respectively. The surface profiles are given with respect to the respective datum of each profile as a positive value, although the height of two profiles is measured in opposite directions. The accuracy of this method can be checked after the surface profiles have been determined using the epoxy resin method. This method eliminates the need to replicate the aperture, but the two surfaces of the fracture must be scanned separately.

5.8 Summary

The fracture flow tests performed in the laboratory were described in this chapter. They were carried out in a high pressure triaxial apparatus using samples of real rock fracture with a sub-axial fracture induced in the laboratory. The steady flow rates

calculated for a particular confining stress and inlet pressure were recorded and the changing confining pressure and inlet pressure, and the flow rates of water for each case were calculated. The fracture aperture distribution in 2D space was determined by replicating the fractured aperture using a fast setting silicon rubber solution and scanning the surfaces using a non-contact 3D laser scanner. The aperture data that was processed using computer software and the calculated flow rates were used in the computer program developed in Chapter 4 to solve the mathematical model derived in Chapter 2. The results obtained results the tests are shown later in the Results and Discussion chapter.

Chapter 6 Results and Discussion

6.1 Introduction

In Chapter 3, the development of a mathematical model and its numerical solution was presented. Chapter 4 described the developed computer programme while Chapter 5 explained the set up and procedures for laboratory testing. In this chapter the mathematical model (Chapter 3) will be verified by comparing the results of the laboratory experiments with the model predictions using the computer programme (Chapter 4). The use of this computer programme in rock fracture flow and deformation predictions will also be discussed.

Four 54 mm diameter specimens of sandstone were split as described in Chapter 5 and artificially fractured specimens were prepared. Two of them, called 'MG' and 'P' were kept as mated fracture specimens while the other two called 'N' and 'SE' were displaced by 3 mm. Both ends of these specimens were trimmed to make dislocated fracture specimens. Their surfaces were scanned and the volumetric flow rate through the fracture was measured in the triaxial cell. The initial deformation data was used to measure the initial joint normal stiffness, and when the flow tests ended, the aperture distribution of the fracture was also measured. The

relevant data were fed into the computer programme and the results of the flow simulation were obtained.

6.2 Results of Laboratory Fracture Flow Test

Flow through rock fractures for multiple hydraulic gradients at increasing confining pressures was studied, and it was noted that fracture flow was extremely sensitive to the aperture, and the fracture aperture was randomly distributed in 2D space. Moreover, the fractures deformed under the applied effective stress, and therefore, the fracture flow rate against effective stress reflected the aperture deformation against effective stress. The fracture flow rates for the mated and dislocated joints that were measured in the laboratory at different confining pressures are as follows.

6.2.1 *Mated Joints*

The surfaces of mated joints were closer everywhere, which resulted in higher initial joint normal stiffness compared to dislocated joints, while the flow rate against hydraulic gradient was increased linearly. Generally, higher flow rates were recorded at lower confining pressures, but when the confining pressure was higher, the flow rates gradually reduced, while the gradient of the flow rate versus the hydraulic gradient at elevated confining pressures also decreased. Figure 6.1 contains the flow rate versus hydraulic gradient variation for the mated joint specimen ‘MG’, while Figure 6.2 shows that for the mated joint specimen ‘P’.

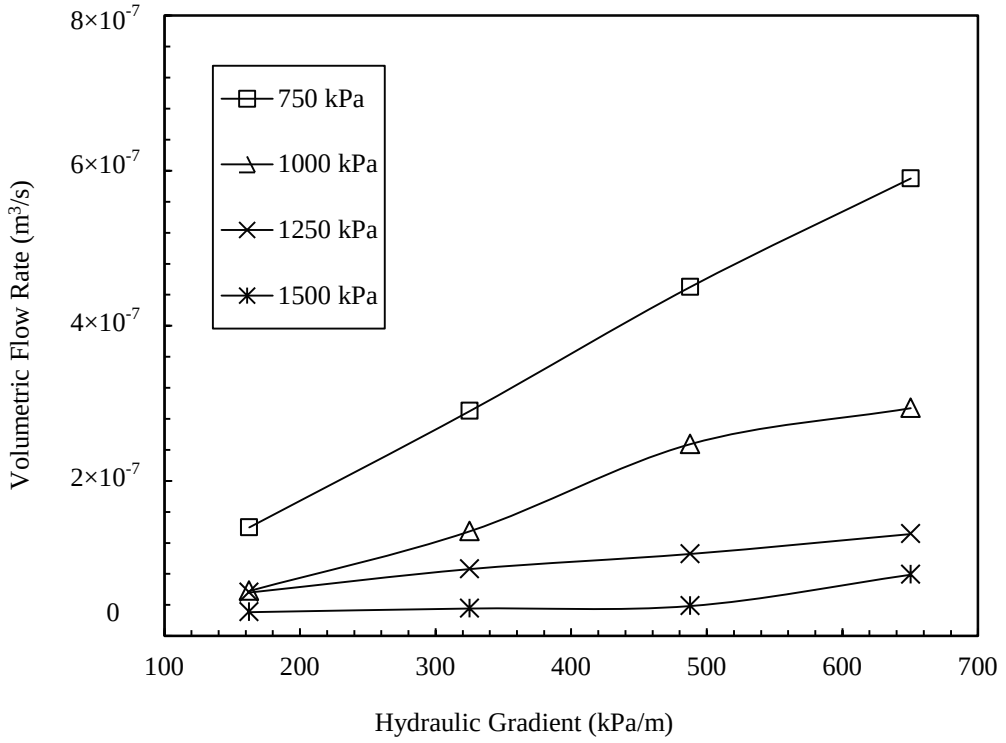


Figure 6.2: Volumetric flow rates recorded at different confining pressures for specimen 'P'

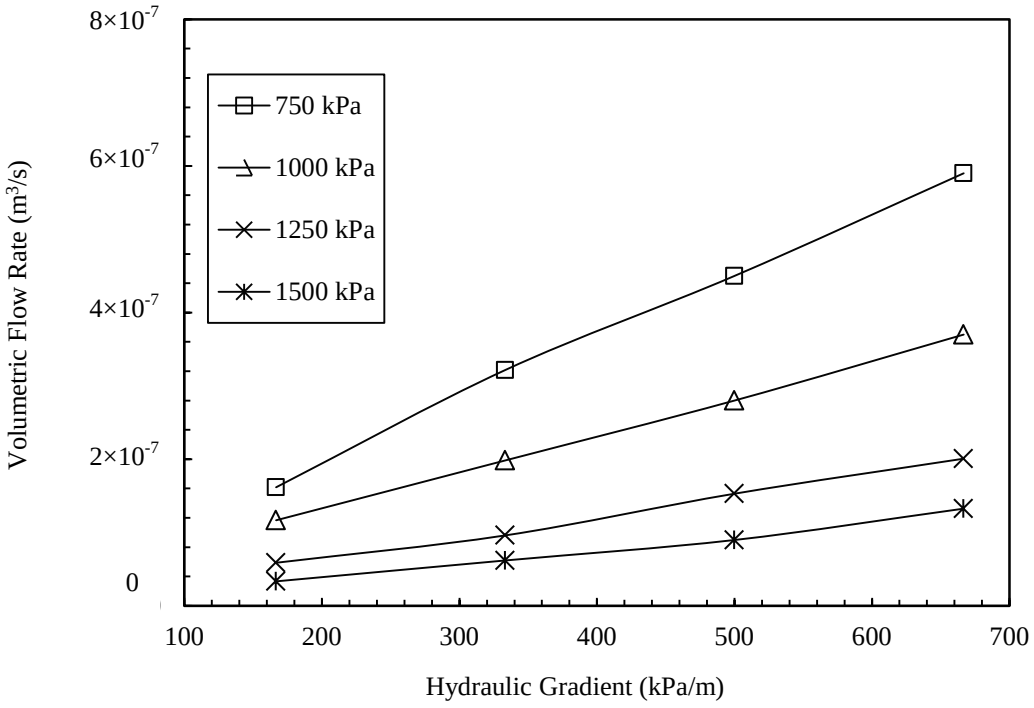


Figure 6.1: Volumetric flow rates recorded at different confining pressures for specimen 'MG'

6.2.2 *Dislocated Joints*

When the fracture surfaces dislocated away from each a completely different aperture distribution resulted. The dislocated joint apertures were usually distributed over a wide range compared to the mated joints. The number of close asperities were less, which caused the initial joint normal stiffness to be lower than the mated joints. The recorded flow rates were higher than the mated joints because there were more open apertures available. Other than that, similar trends for the flow rate against the hydraulic gradient were observed for dislocated joint apertures too. For the specimens 'N' and 'SE', the flow rate and hydraulic gradient plots for different confining pressures are shown in Figures 6.3 and 6.4, respectively.

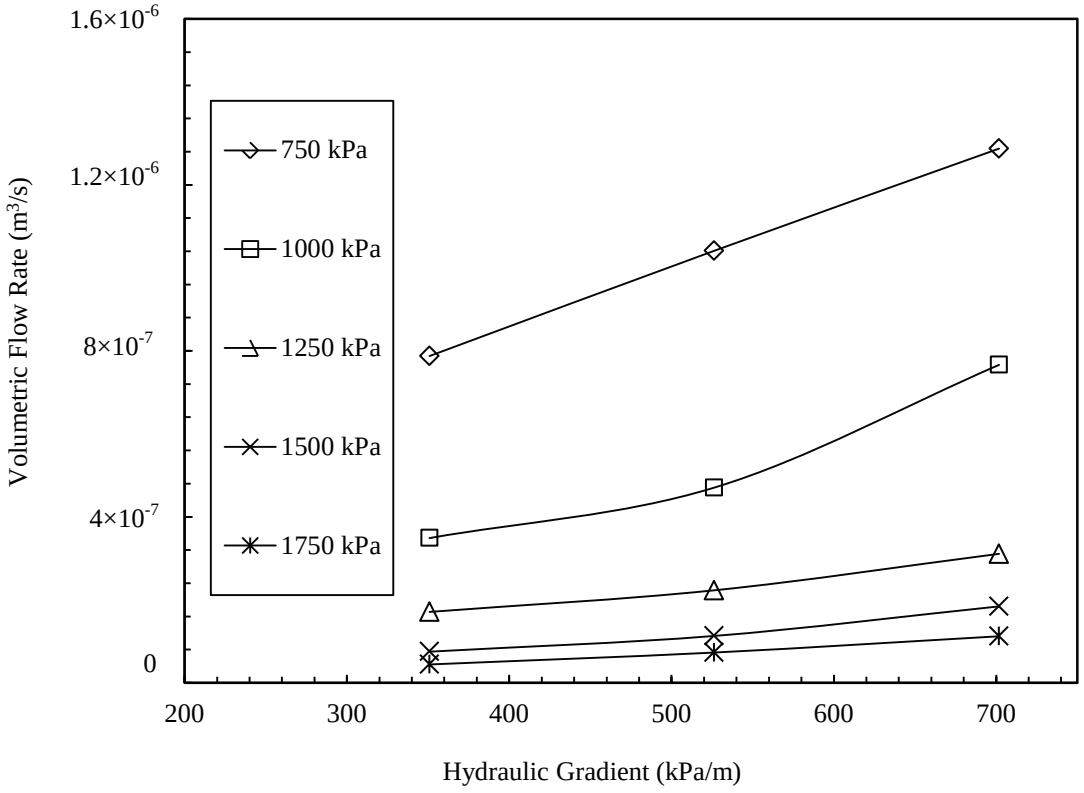


Figure 6.3: Volumetric flow rates recorded at different confining pressures for specimen 'N'

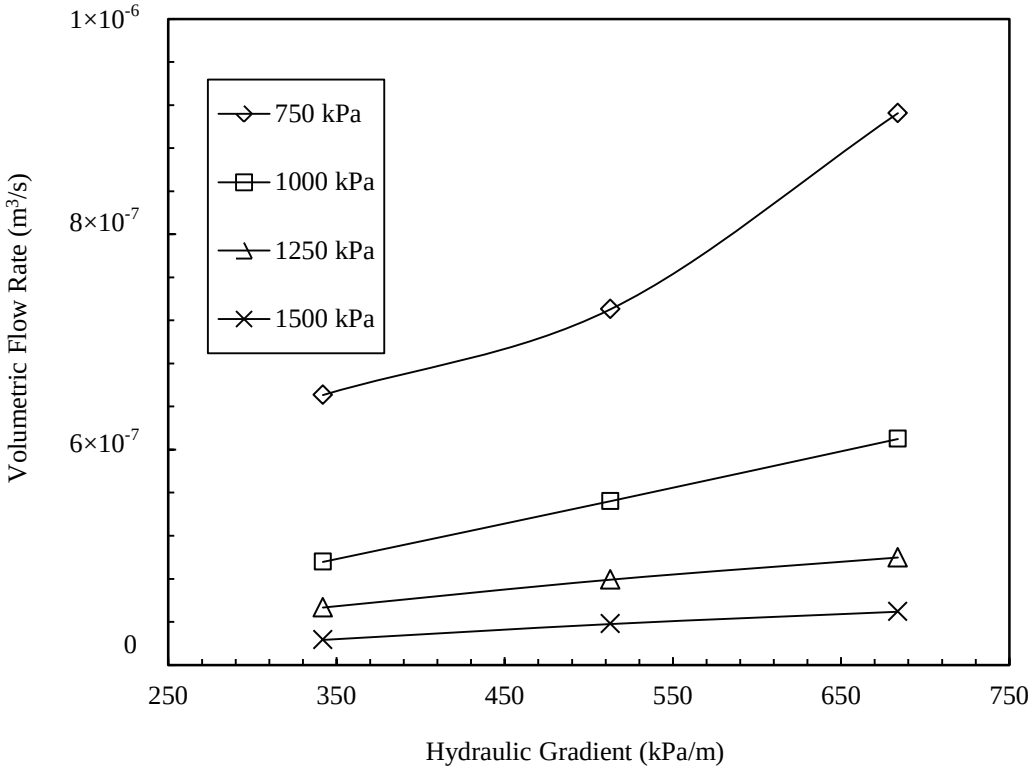


Figure 6.4: Volumetric flow rates recorded at different confining pressures for specimen 'SE'

6.3 Rock Fracture Surface Scanning and Aperture Measurements

The procedure for measuring the aperture of fractures was described in Chapter 5, while the generation of a fracture surface profile and measuring the fracture aperture of the sandstone specimens used in this study are presented here. Fracture surfaces were scanned before the flow tests were conducted to check for any asperity damage after the tests. Since no shear movements were applied and moderate confining pressure range was applied, no asperity damage was observed. The surface profile data were used to calculate the Joint Roughness Coefficient (JRC) using the statistical relationship (Tse and Cruden 1979) given in Equation (3.36).

$$JRC = 32.2 + 32.44\log(z_2) \quad (6.1)$$

In the above,

$z_2 = \left(\frac{1}{L_n}\sum_{i=1}^{N_p-1}\frac{(z_{i+1}-z_i)^2}{(x_{i+1}-x_i)}\right)^{1/2}$ is the root mean square of the first derivative of the profile given in discrete form, L_n is the length of the profile in the x-direction, N_p is the number of discrete points in the x-direction, and z_i and x_i are the elevation and length of a discrete point in the x-direction respectively. The average value of the JRC was taken from the linear profiles that were equally spaced and parallel to each other on the surface, while the average JRC values for the four specimens are given in Table 6.1. JRC ranges from 0 to 20 with smooth surfaces having a lower JRC while rough surfaces having JRC closer to 20. The joint surfaces used in this study were moderately rough surfaces according to Table 6.1.

Table 6.1: Average Joint Roughness Coefficients of the specimens

Specimen	MG	P	N	SE
Average JRC	5.4	6.5	5.9	7.4

To calculate the JRC of the surface, five profiles were selected on the surface which 10 mm apart from each other and profiles were plotted in the following figures. The surface profiles of the mated joint specimens ‘MG’ and ‘P’ are shown in Figures 6.5 and 6.6, respectively. Figures 6.7 and 6.8 contain the surface profiles of the dislocated joint specimens ‘SE’ and ‘N’, respectively.

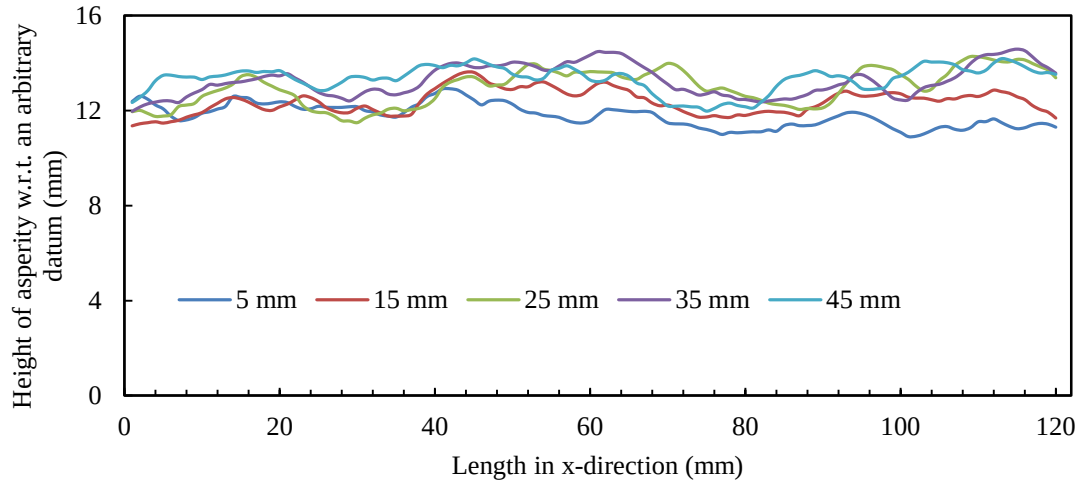


Figure 6.5: Roughness profiles of specimen 'MG' at different lengths from the edge

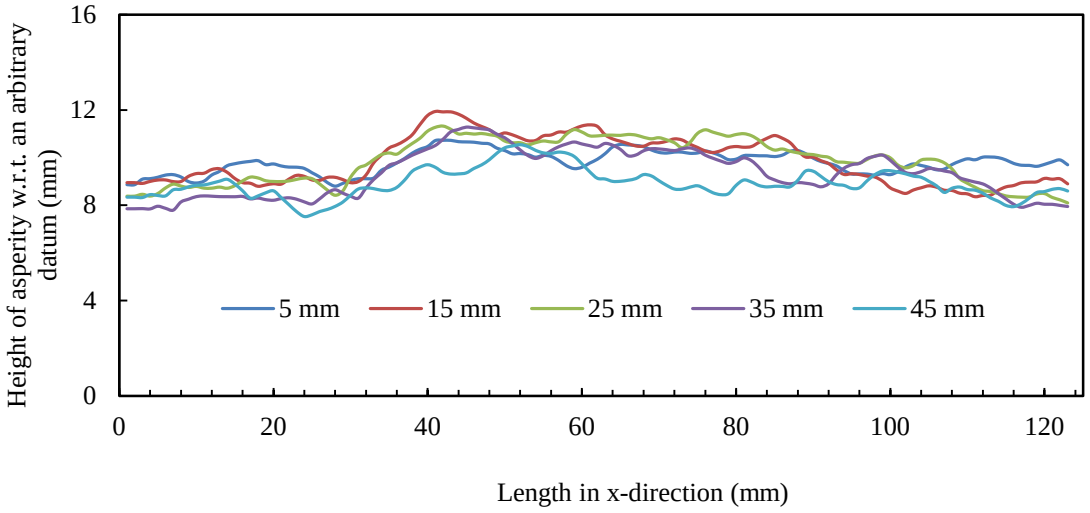


Figure 6.6: Roughness profiles of specimen 'P' at different lengths from the edge

At the end of the triaxial test, the fracture apertures of the specimen were measured using the method explained in Chapter 5, while the initial aperture distributions of the tested fractures are shown in the following figures. The mated

joint apertures were smaller than the dislocated joint apertures and their distribution was limited, whereas the apertures of the dislocated joints had a wider ranging span.

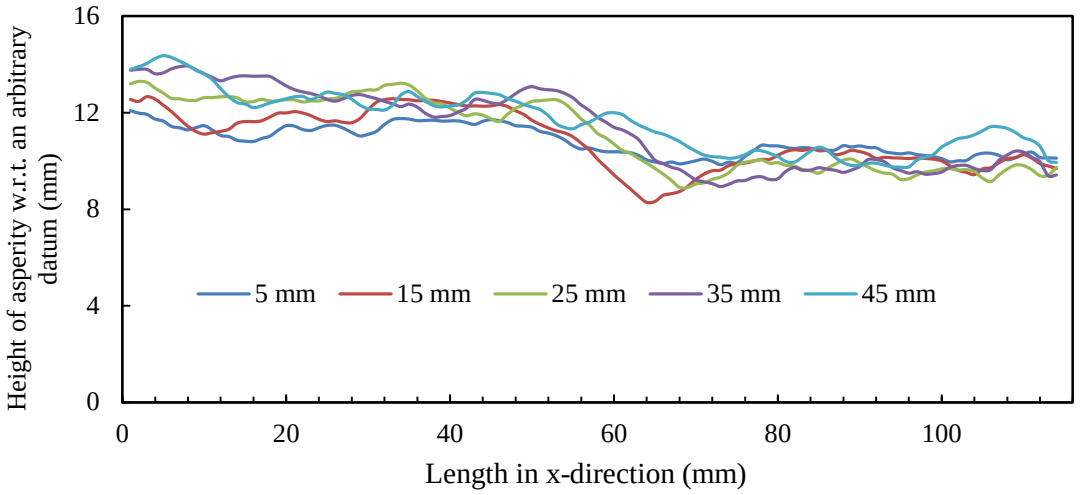


Figure 6.7: Roughness profiles of specimen 'N' at different lengths from the edge

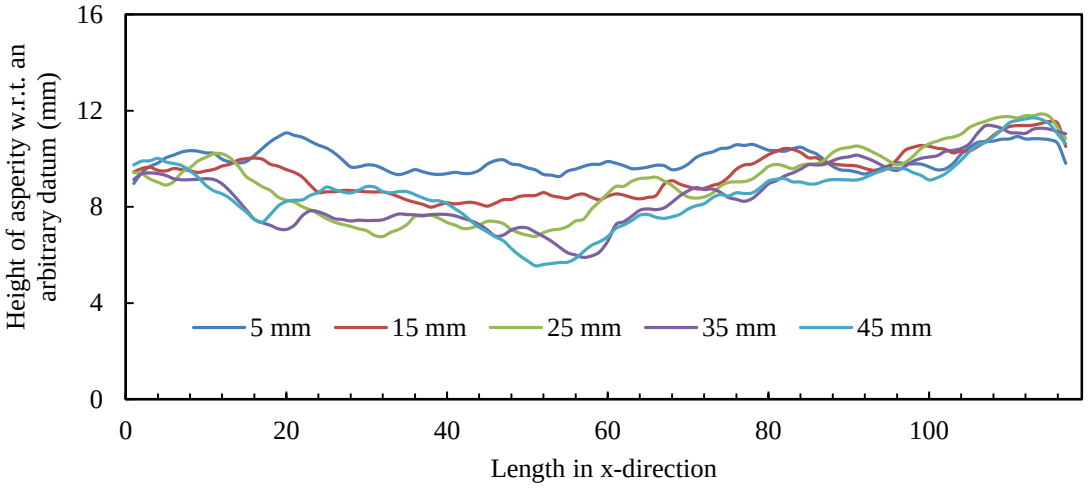


Figure 6.8: Roughness profiles of specimen 'SE' at different lengths from the edge

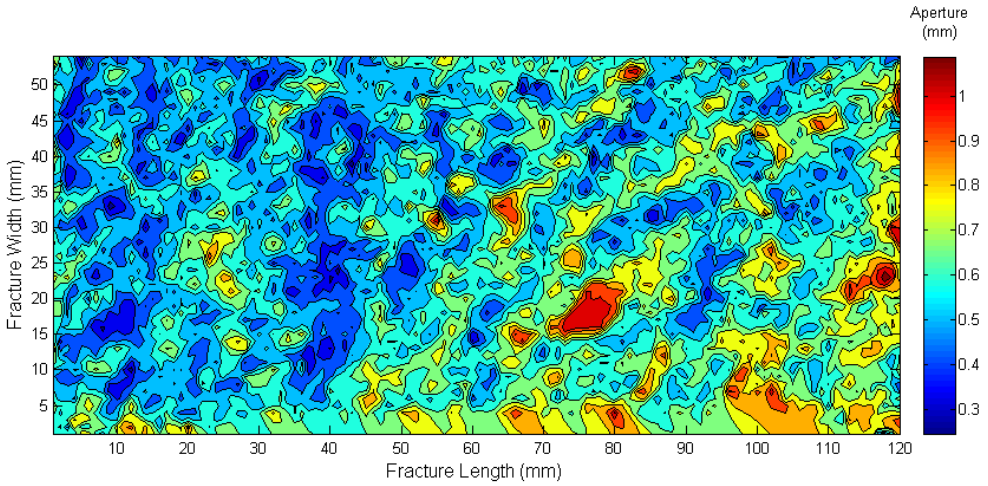


Figure 6.9: Initial aperture of mated specimen 'MG'

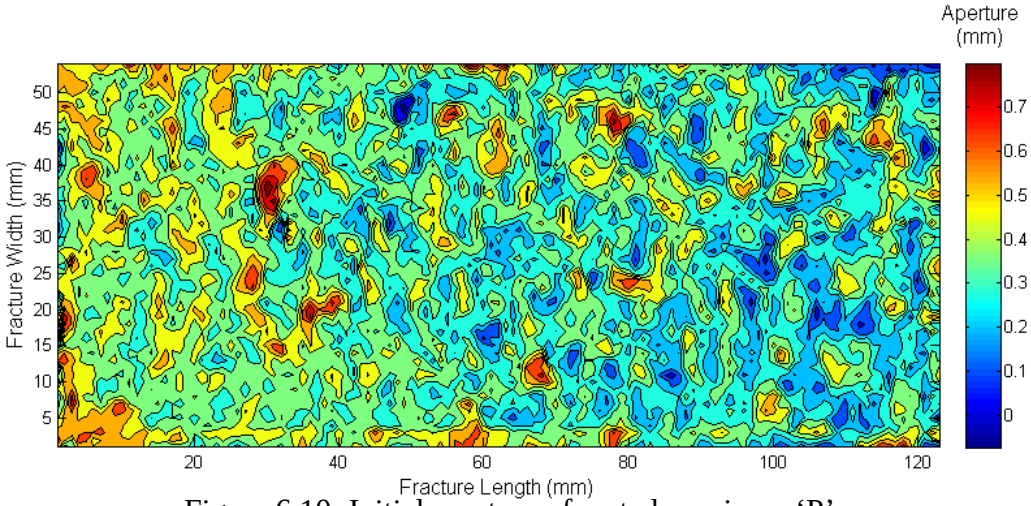


Figure 6.10: Initial aperture of mated specimen 'P'

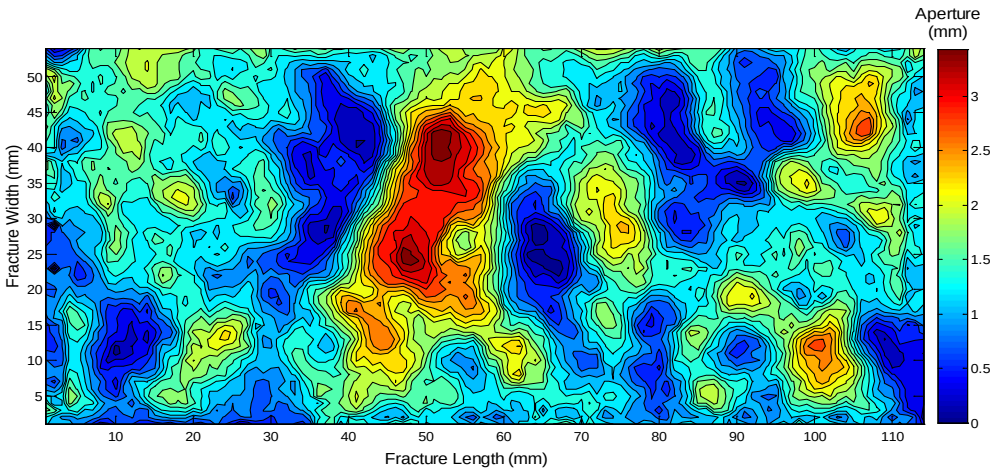


Figure 6.11: Initial aperture of dislocated specimen 'N'

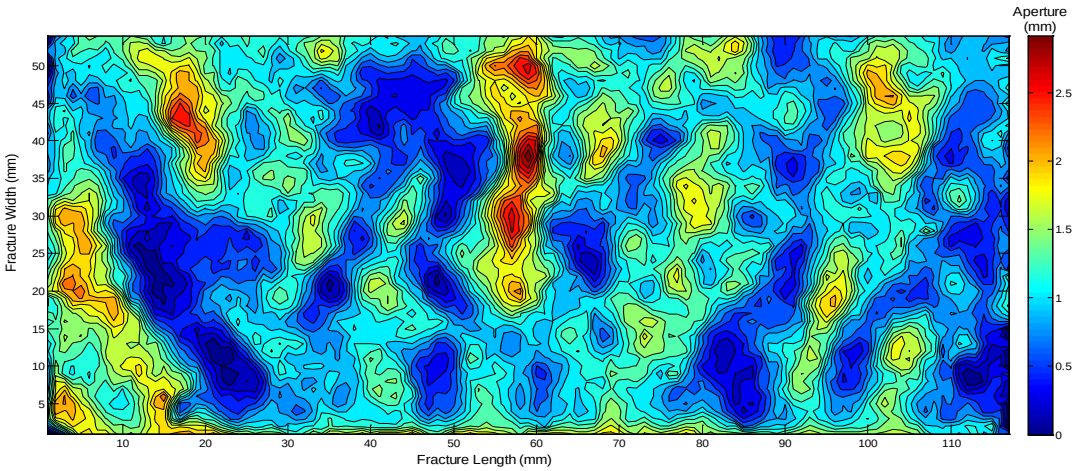


Figure 6.12: Initial aperture of dislocated specimen 'SE'

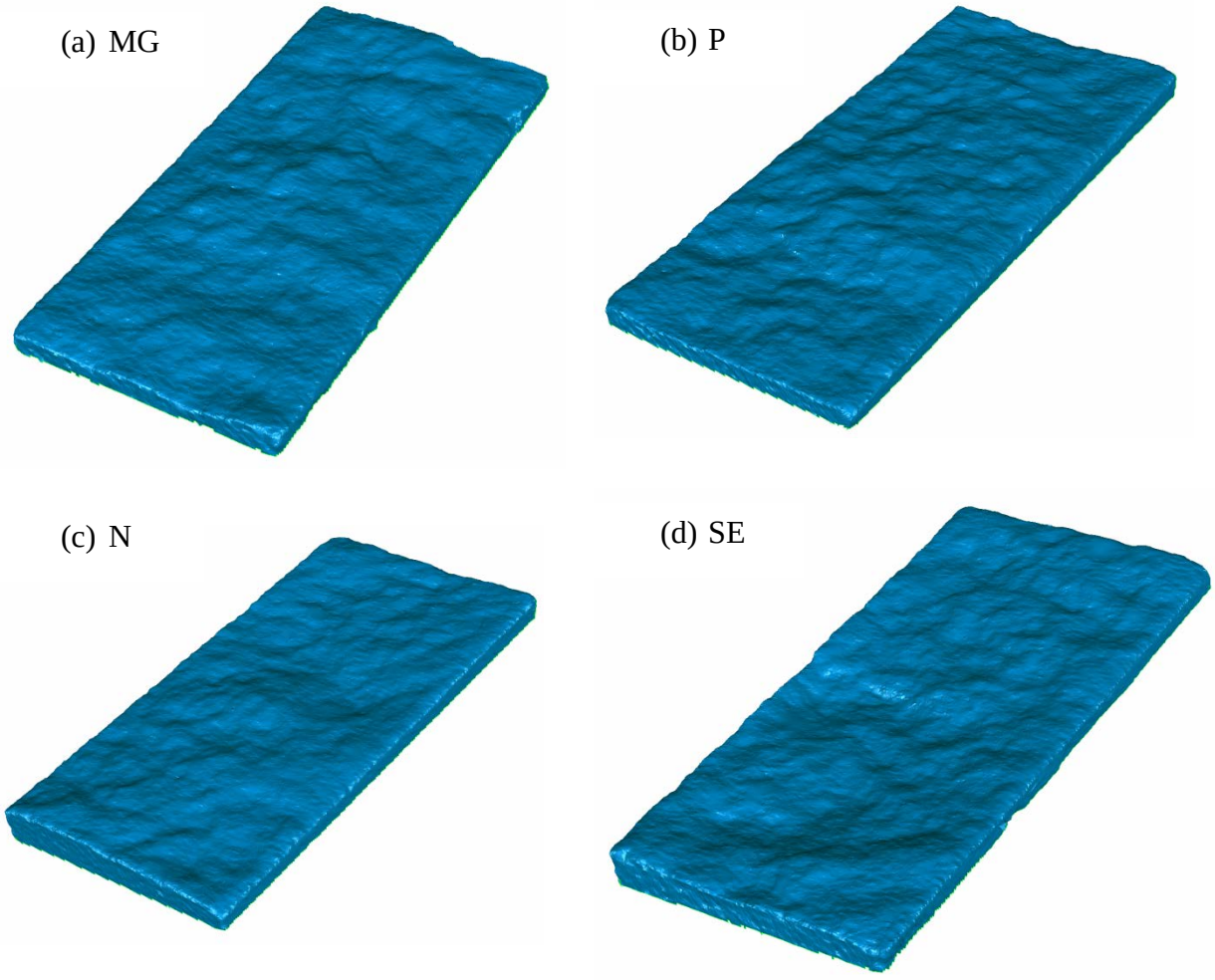


Figure 6.13: Scan images of the specimen surfaces

Figures 6.9 and 6.10 are the contour plots of the mated joints 'MG' and 'P' respectively, while the contour plots for the dislocated joints 'N' and 'SE' are shown in Figures 6.11 and 6.12, respectively. Scan images obtained from the 3D laser scanner are grouped in Figure 6.13.

6.4 Numerical Model Convergence

Having a numerical solution of partial differential equation converge closer to the exact solution is an essential need in numerical analysis because by default, a numerical solution is only an approximation. The first order partial derivative of a function is mathematically expressed as follows;

$$\frac{\partial f(x_1,..x_i..)}{\partial x_i} = \lim_{\Delta x_i \to 0} \left(\frac{f(x_1,..(x_i + \Delta x_i)..) - f(x_1,..x_i..)}{\Delta x_i} \right) \tag{6.2}$$

The actual solution can be found when Δx_i or the mesh size of a finite volume model is close to zero, so when the mesh size is reduced, the numerical solution should converge to a certain value. This can be checked by reducing the mesh size to a finer value and then selecting a size which can give a solution closer to the actual solution. Using much finer grids can make the solution time quite long, which will not be effective for certain applications, so in this study, convergence of the numerical model was checked for actual aperture distribution, beginning with a much coarser grid and reducing it to a finer grid, and then recording the output flow rates.

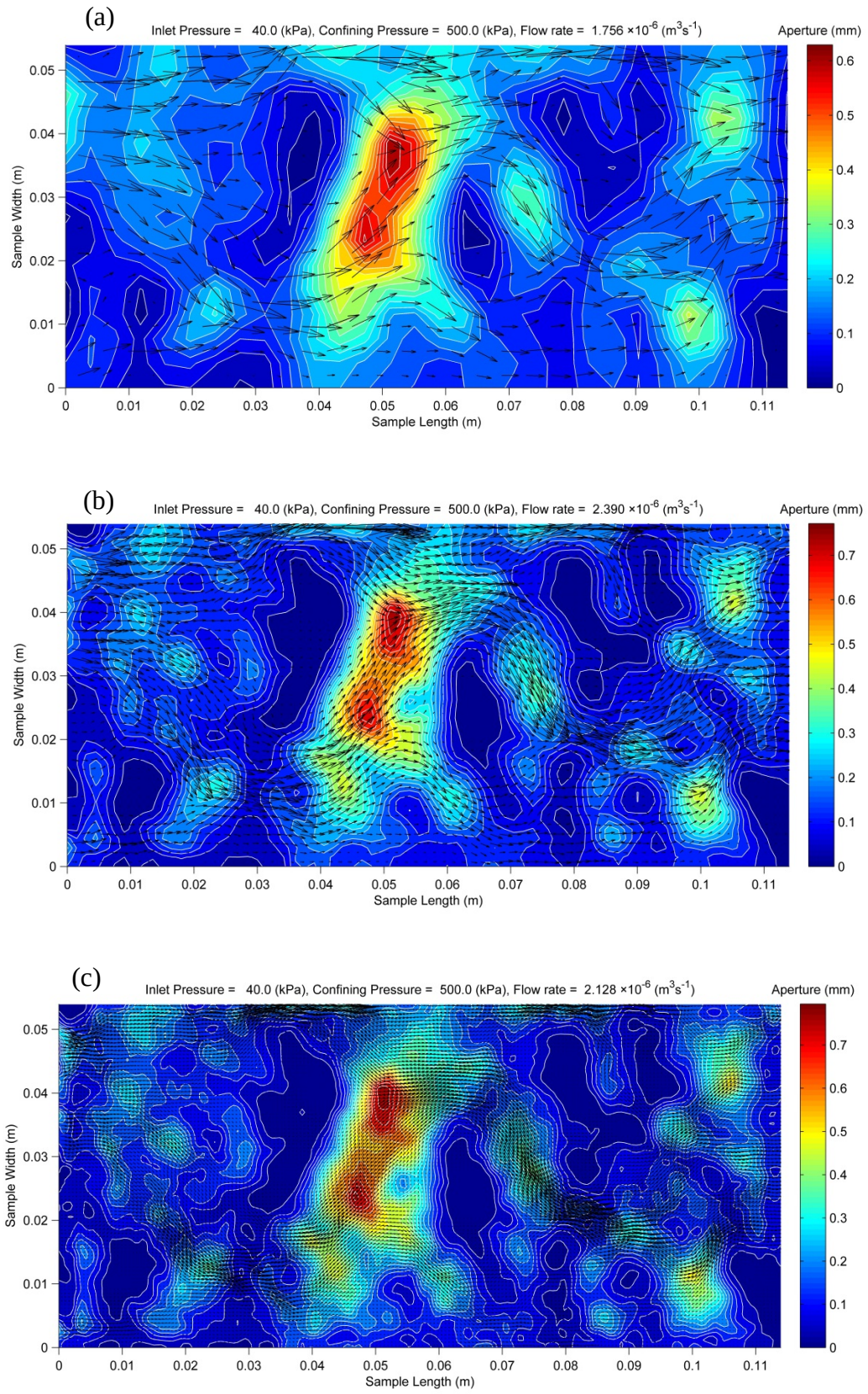


Figure 6.14: Flow simulations with 3 different grid sizes. (a) 4 mm (b) 1 mm (c) 0.25 mm

Figure 6.14 shows three instances of the flow simulation for specimen 'N' while the grid size of the flow domain was reduced. Figure 6.15 shows the convergence of the solution with the grid size. Note that the x-axis of this graph is decreasing to the right and the flow rate was converging after 1mm size grid. For finer grids, the calculation time increased and velocity vector plot arrows were overcrowded. This can be fixed by selectively plotting the vectors, although the programme is currently not configured for that. Convergence of the numerical solution was confirmed and 1 mm grid size was chosen for this study to simulate the fracture flows.

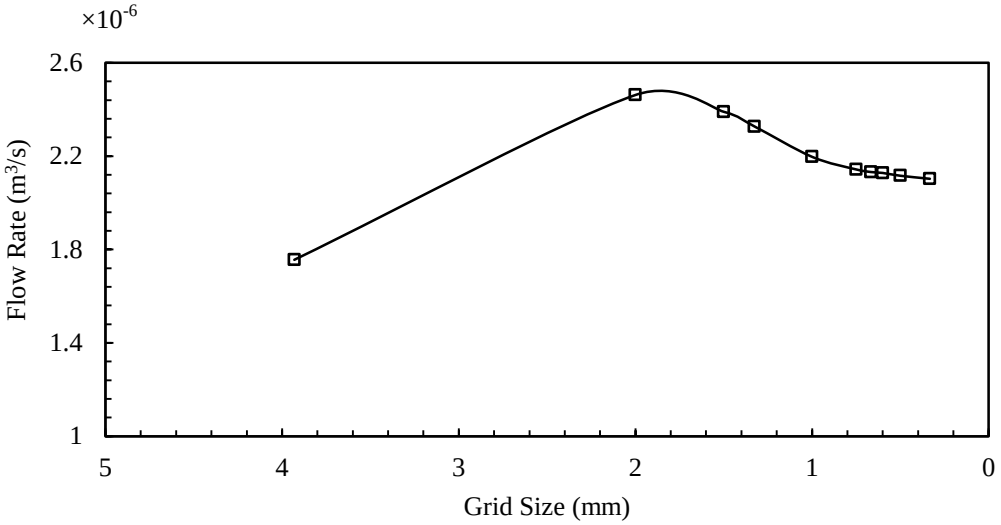


Figure 6.15: Flow rate convergence with grid size reduction for specimen 'N'

6.5 Model Prediction and Verification

The measured apertures were entered into the computer program together with the other properties shown in the Table 6.2. The programme was then executed for the four specimens for confining pressure in the range 750 kPa to 1750 kPa and different

inlet pressures of 40 kPa, 60 kPa and 80 kPa. In the laboratory tests, the confining pressure was increased by intervals of 250 kPa, although the test programme could run for lower increments in order to obtain a smooth relationship. Simulations were observed for aperture deformation and flow behaviour with confining pressure increments of 50 kPa, but only four confining pressure stages for one specimen are displayed here.

Table 6.2: Input variables to the computer programme

Specimen	MG	P	N	SE
Specimen length (mm)	120	123	114	117
Specimen width (mm)	54	54	54	54
Initial joint normal stiffness (Pa/m)	8.2×10^{7}	3.1×10^{8}	3.8×10^{7}	4.8×10^{7}
Fluid density	1000 $kg.m^{-3}$			
Fluid viscosity	0.001 Pa.s			
Acceleration of gravity	9.81 ms^{-2}			

The undulating nature of the aperture resulted in a non-uniform pressure distribution, which meant the correct pressure had to be calculated in order to define the direction of flow at a particular control volume of the flow domain. The inlet and out let boundaries were kept at constant pressures and according to that boundary condition, the pressure equations were solved in the programme, as described in Chapter 3. The contour plots of the pressure distribution for the four specimens are shown in Figures 6.16, 6.17, 6.18 and 6.19 for specimens ‘MG’, ‘P’, ‘N’ and ‘SE’, respectively. These figures show that the pressure distribution for dislocated joints

was uneven because they contained areas with large apertures as well as areas with smaller apertures, whereas the mated joint apertures did not show excessive variance, the pressure distribution was less uneven.

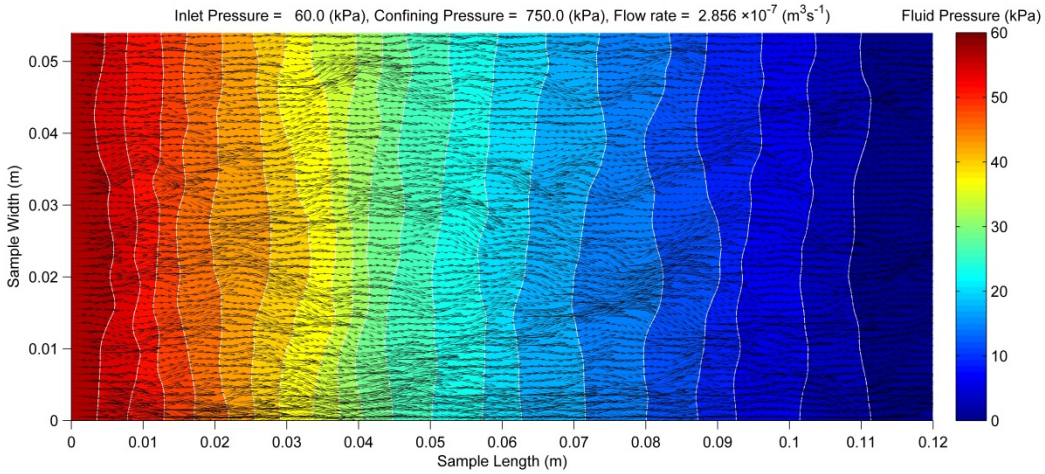


Figure 6.16: Flow simulations of specimen 'MG' with pressure distribution for 60 kPa inlet pressure and 750 kPa confining pressure

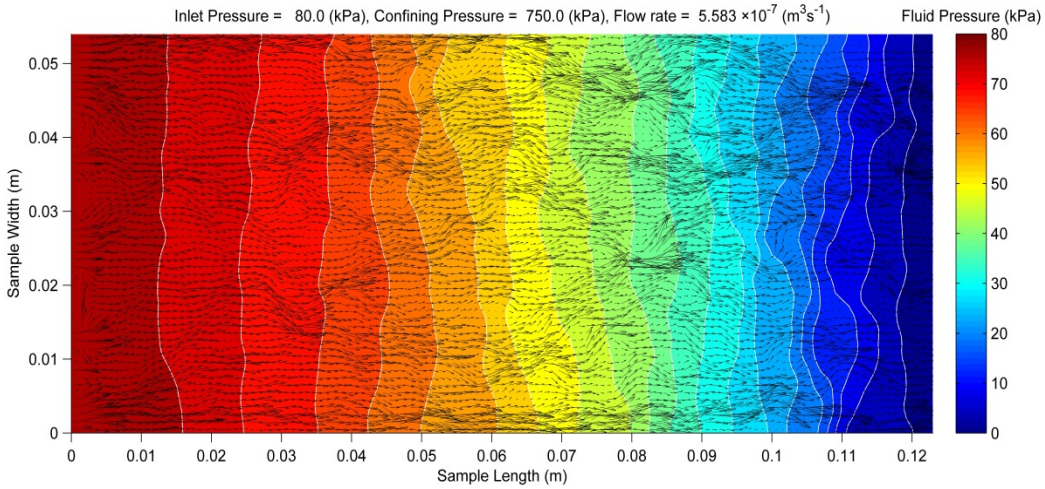


Figure 6.17: Flow simulations of specimen 'P' with pressure distribution for 80 kPa inlet pressure and 750 kPa confining pressure

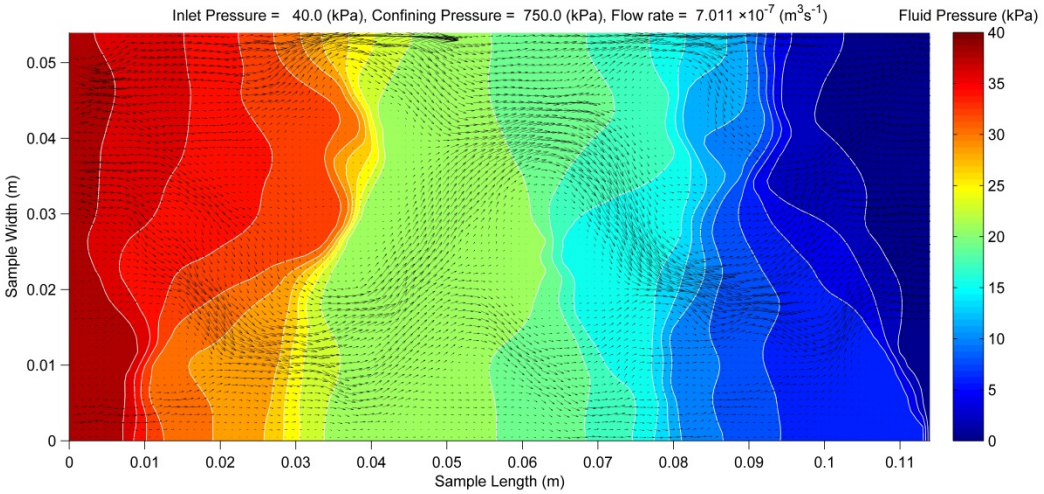


Figure 6.18: Flow simulations of specimen 'N' with pressure distribution for 40 kPa inlet pressure and 750 kPa confining pressure

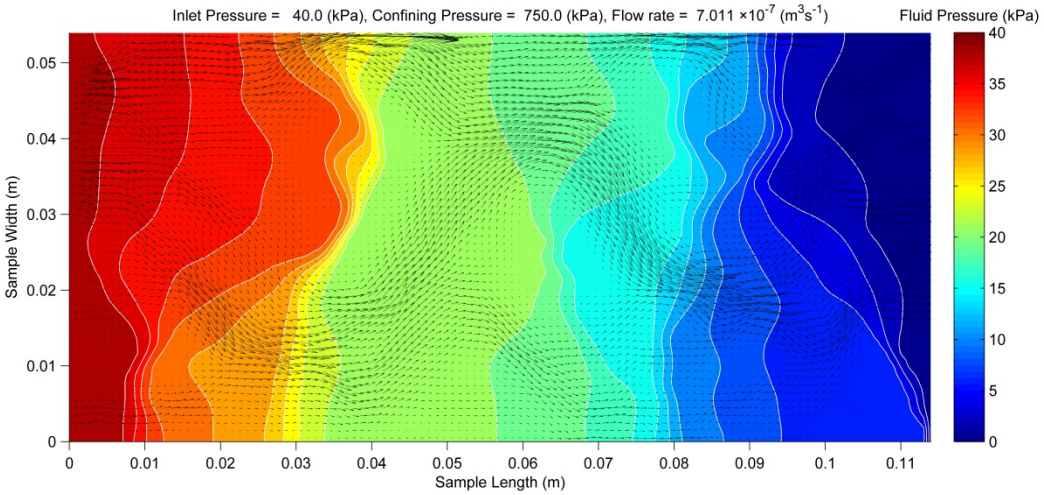


Figure 6.19: Flow simulations of specimen 'SE' with pressure distribution for 40 kPa inlet pressure and 750 kPa confining pressure

The flow behaviour of the fracture is discussed next using velocity vector plots. The aperture distribution is plotted as a contour map to compare with flow paths.

6.5.1 Specimen 'MG'

Figure 6.20 shows the flow rate tested for increasing confining stress together with the model predictions and Reynolds number related to the flow, while Figure 6.21 shows some velocity vector plots with aperture distribution. The flow rates predicted for confining pressures less than 1000 kPa and the inlet water pressures greater than 60 kPa deviated a little from the laboratory results. However, since the apertures were larger at lower confining pressures, higher hydraulic gradients produced faster flows with non-negligible advection effects. Because the model did not calculate the advection effect, this difference may have occurred, however, underground hydraulic gradients are usually lower and the model predictions for other combinations of inlet pressure and confining pressure were acceptable whose Reynolds number is less than 10.

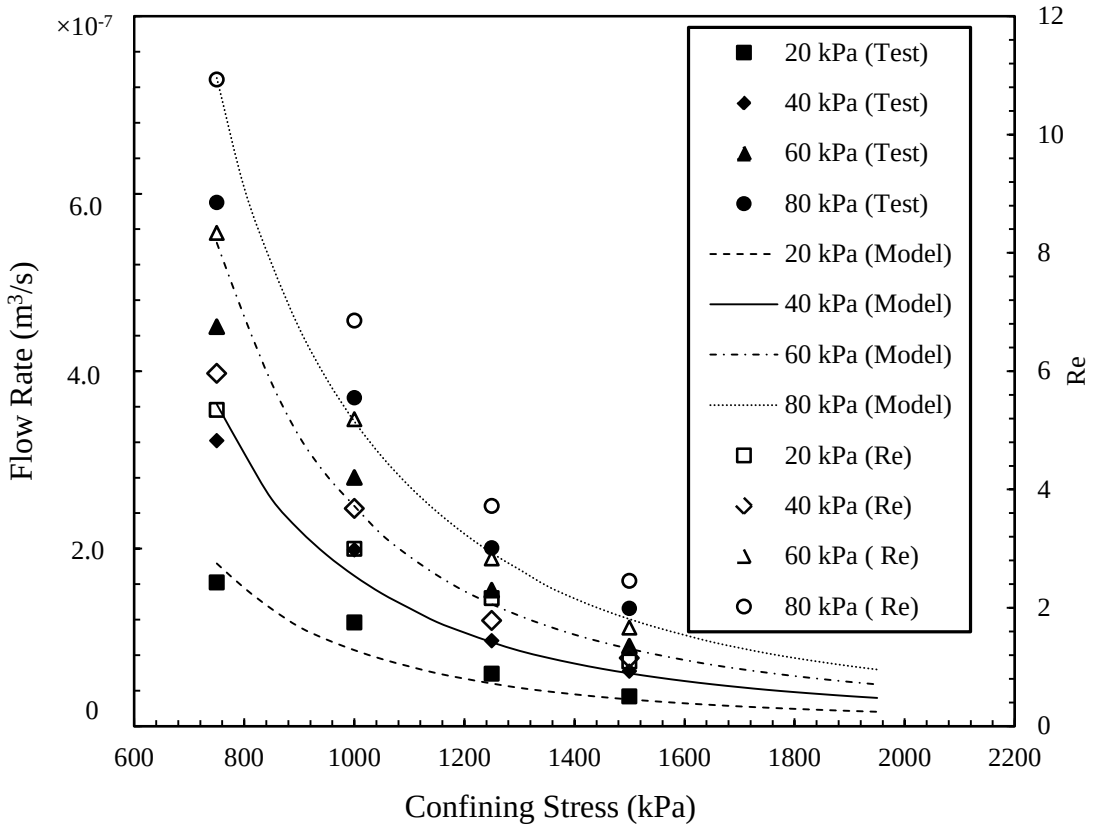
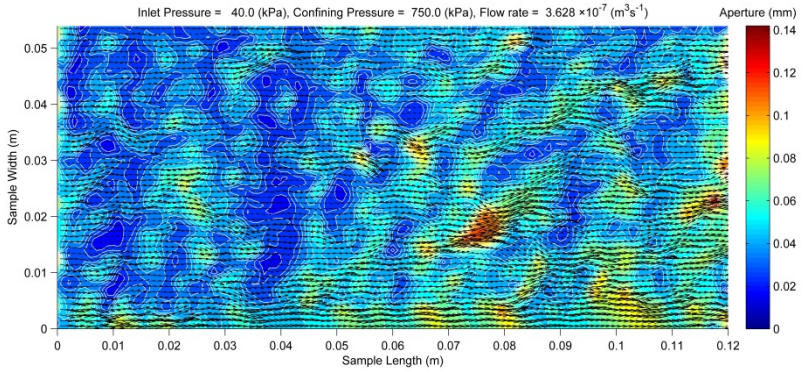
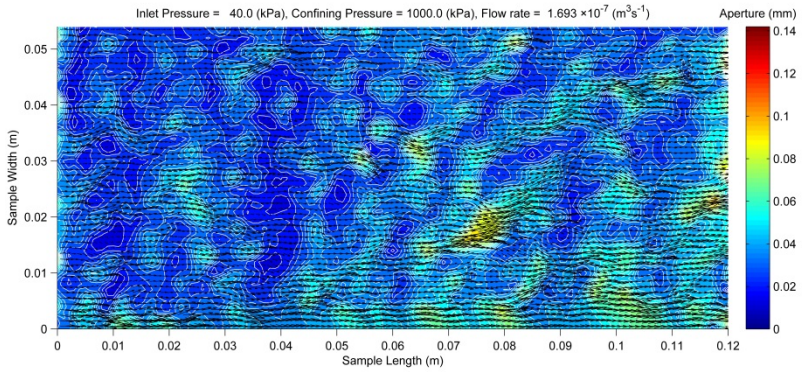


Figure 6.20: Flow rate vs Confining pressure behaviour of specimen 'MG'

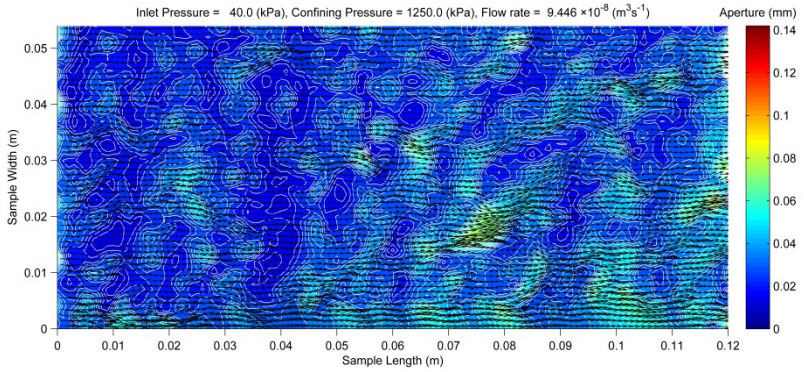
(a)



(b)



(c)



(d)

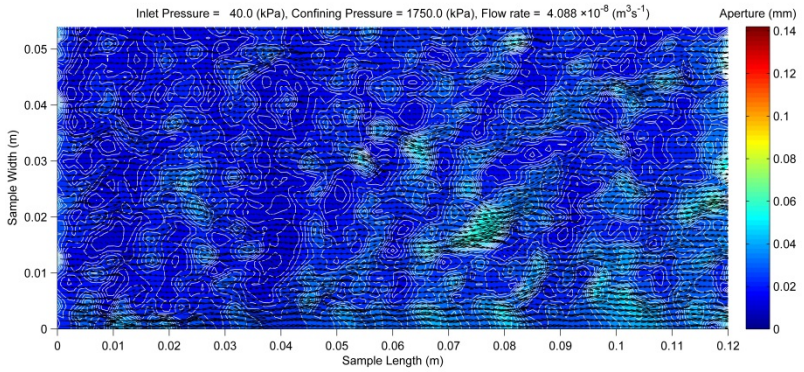


Figure 6.21: Flow simulations of specimen ‘MG’ for increasing confining pressures for 40 kPa inlet pressure. (a) 750 kPa (b) 1000 kPa (c) 1250 kPa (d) 1750 kPa

6.5.2 Specimen 'P'

The flow behaviour against confining stress, together with the Reynolds number for mated joint specimen 'P' is given in Figure 6.22, and the velocity vector plot with aperture distribution for confining stresses of 800 kPa, 1100 kPa, 1400 kPa and 1700 kPa at 60 kPa inlet pressure are shown in Figure 6.23. The behaviour of this was similar to the previous mated specimen where the flow rates were slightly higher because the apertures were larger. The flow rates measured at a confining pressure of less than 1000 kPa and an inlet pressures above 60 kPa were overestimated in the model predictions for the same reason. The model predictions for this specimen agreed with the test results for flow rates having a Reynolds number of less than 10.

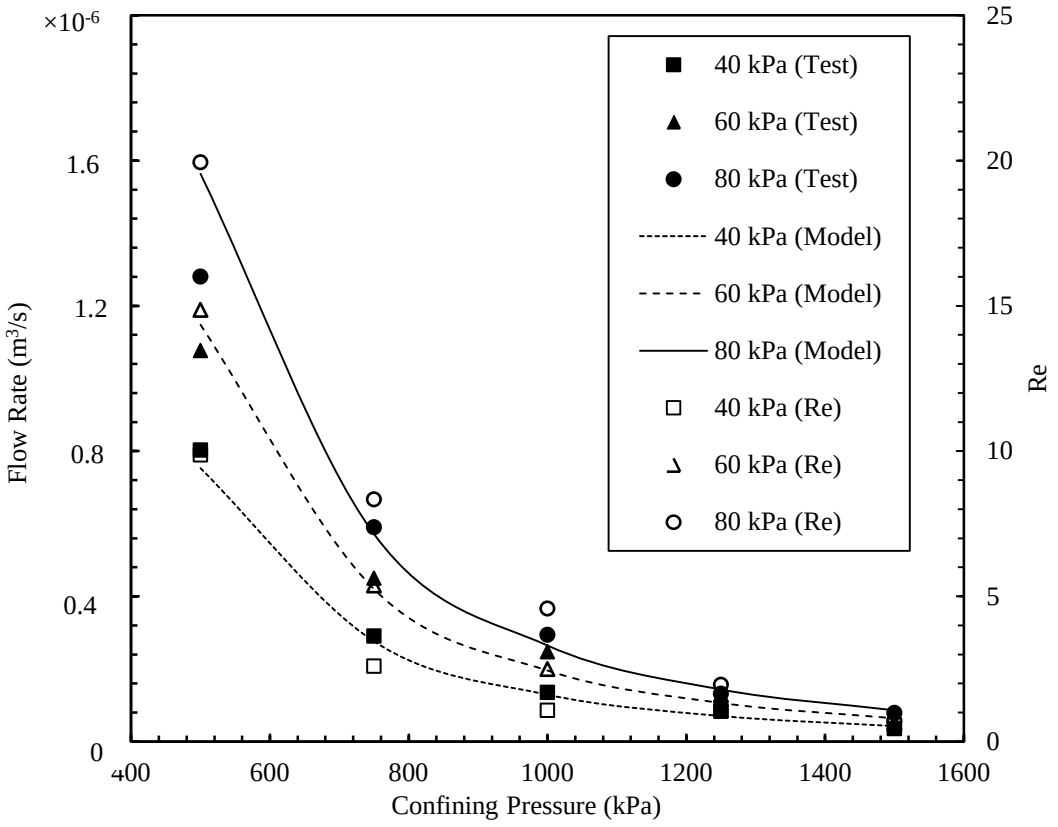
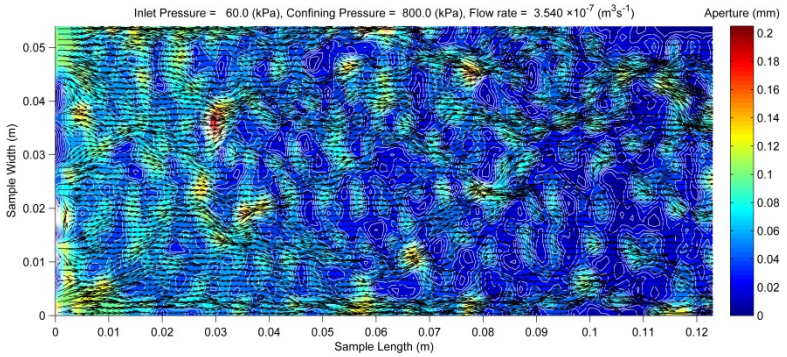
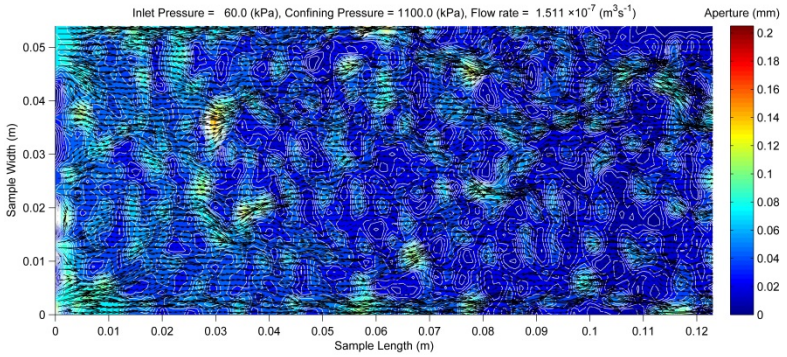


Figure 6.22: Flow rate vs Confining pressure behaviour of specimen 'P'

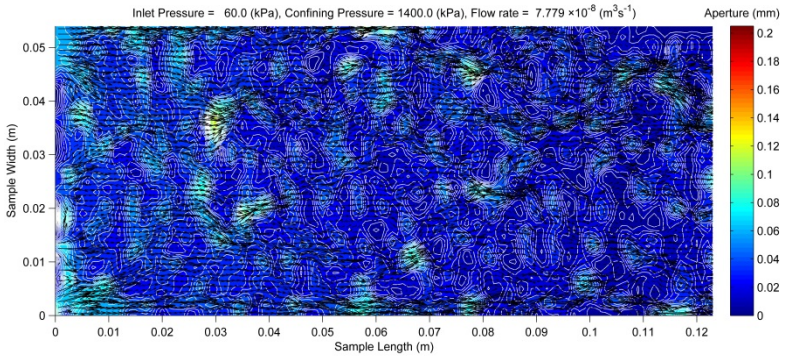
(a)



(b)



(c)



(d)

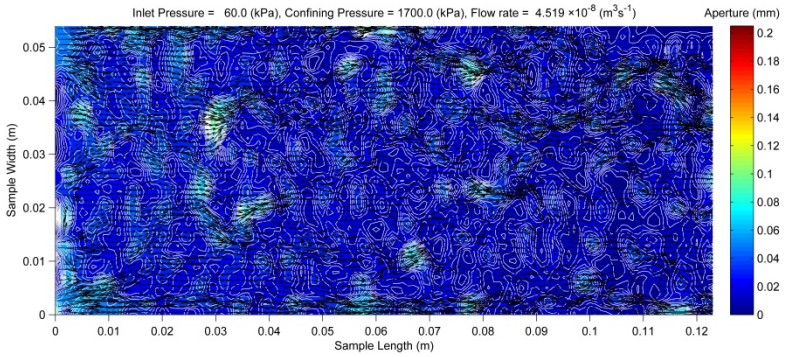


Figure 6.23: Flow simulations of specimen 'P' for increasing confining pressures for 60 kPa inlet pressure. (a) 800 kPa (b) 1100 kPa (c) 1400 kPa (d) 1700 kPa

6.5.3 Specimen 'N'

Figure 6.24 shows the flow rate and Reynolds number versus the confining pressure, while Figure 6.25 shows the velocity vector plot with aperture distribution in the background. The model predictions for dislocated joints tallied with the laboratory experiments for flows with a Reynolds number of less than 10.

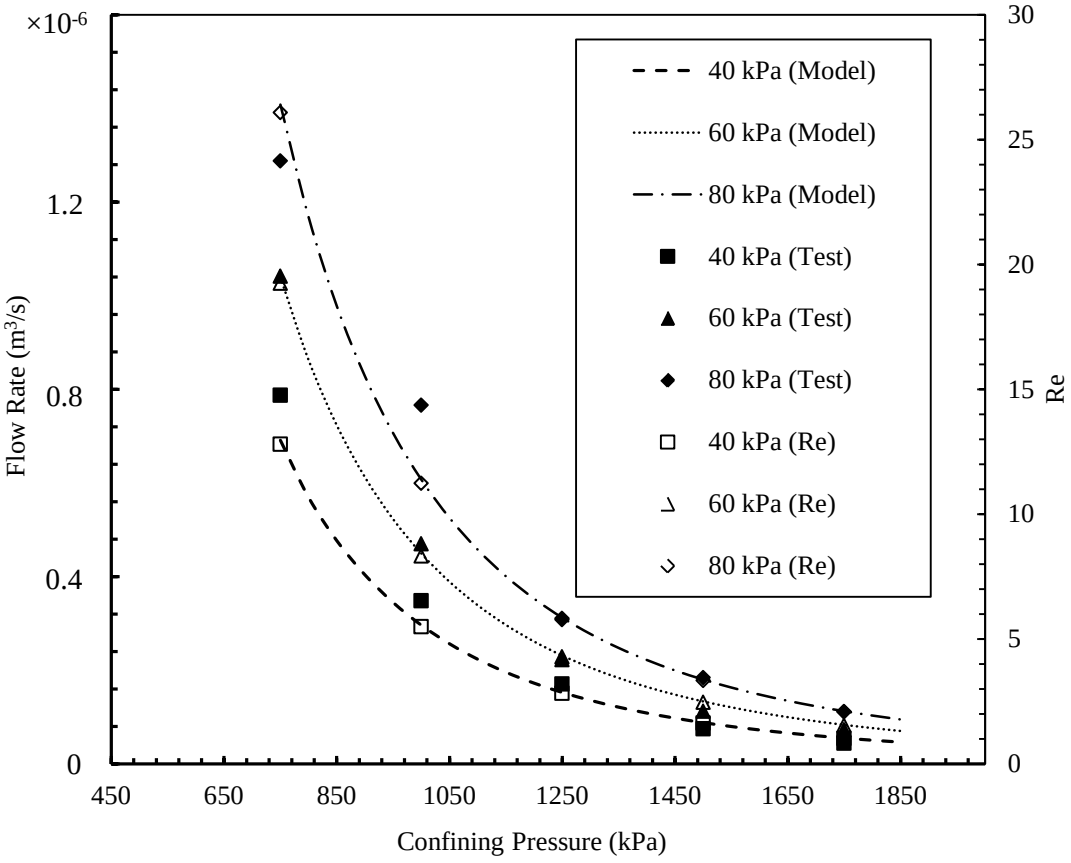
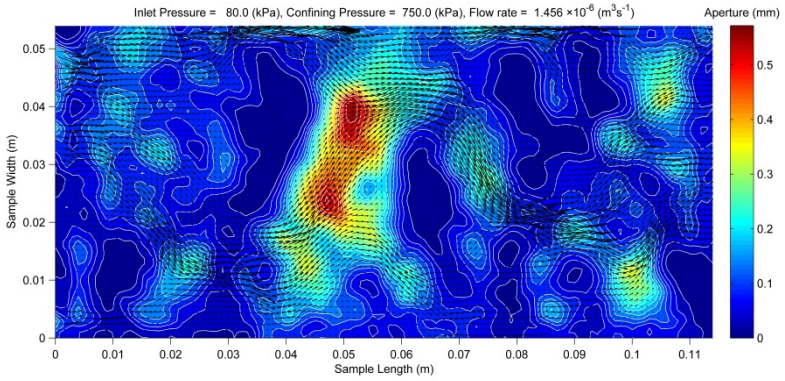
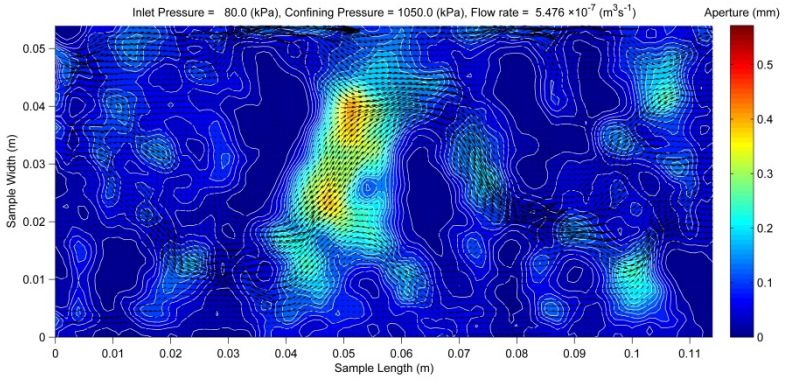


Figure 6.24: Flow rate vs Confining pressure behaviour of specimen 'N'

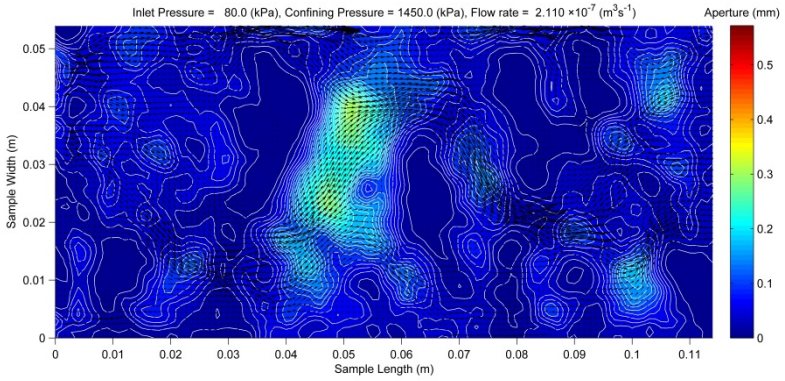
(a)



(b)



(c)



(d)

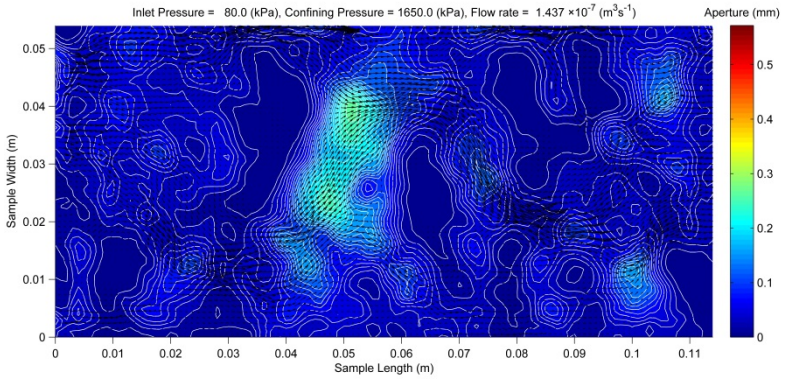


Figure 6.25: Flow simulations of specimen 'N' for increasing confining pressures for 80 kPa inlet pressure. (a) 750 kPa (b) 1050 kPa (c) 1450 kPa (d) 1650 kPa

6.5.4 Specimen ‘SE’

Variations in the flow rate and Reynolds number against the applied confining pressure for specimen ‘SE’ is shown in Figure 6.26. The velocity vector plot with the aperture distribution at four confining pressures for the same specimen, are shown in Figure 6.27. For this dislocated joint specimen too, the Reynolds numbers became higher at lower confining pressures due to higher flow rates, and the model agreement was valid for flows with a Reynolds number of less than 10.

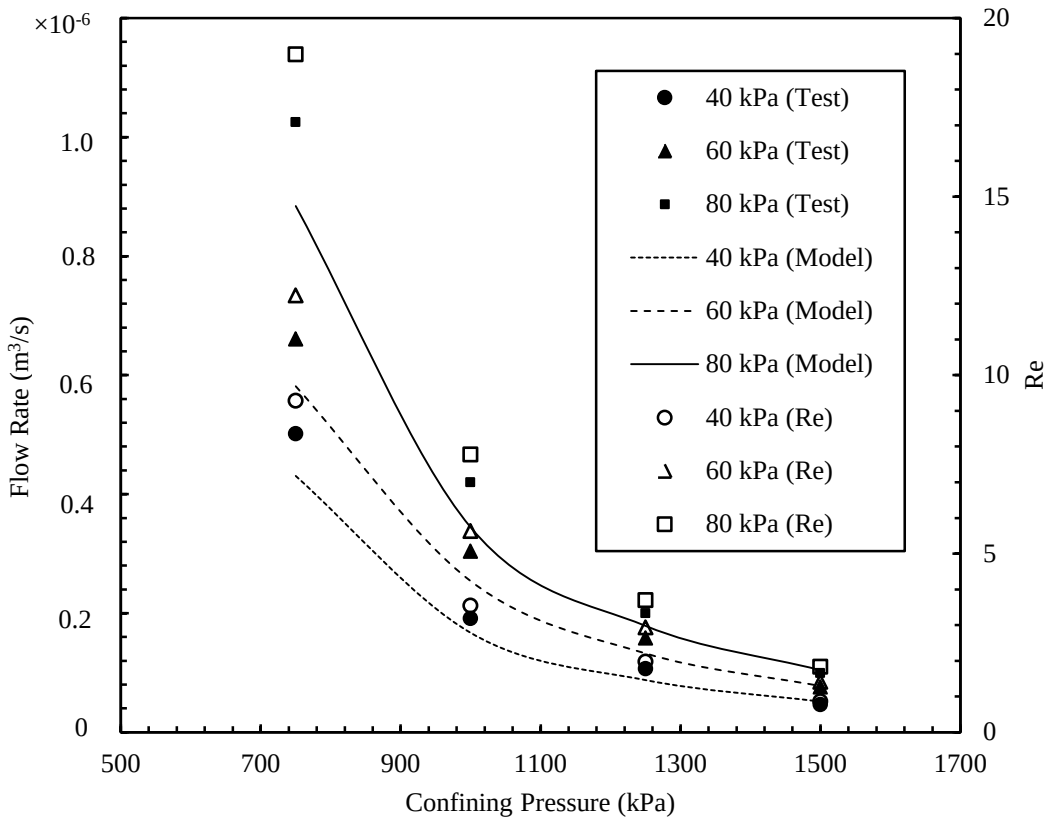
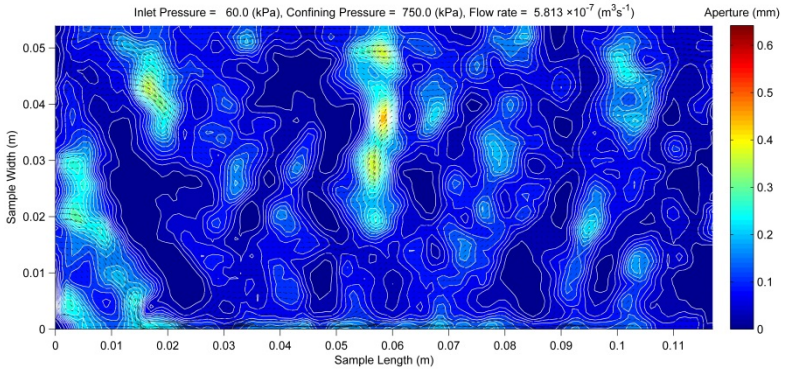
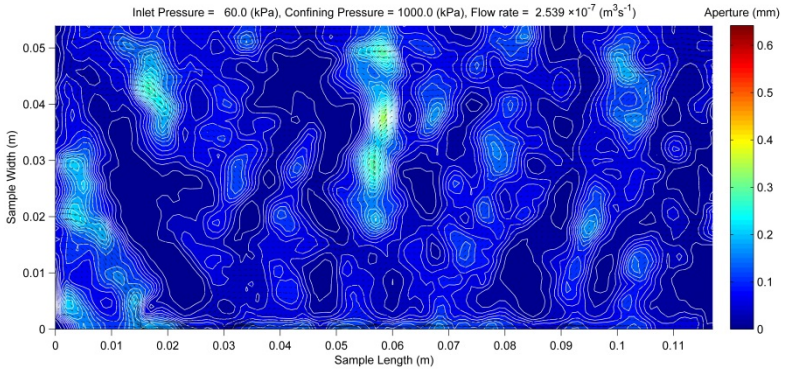


Figure 6.26: Flow rate vs Confining pressure behaviour of specimen ‘SE’

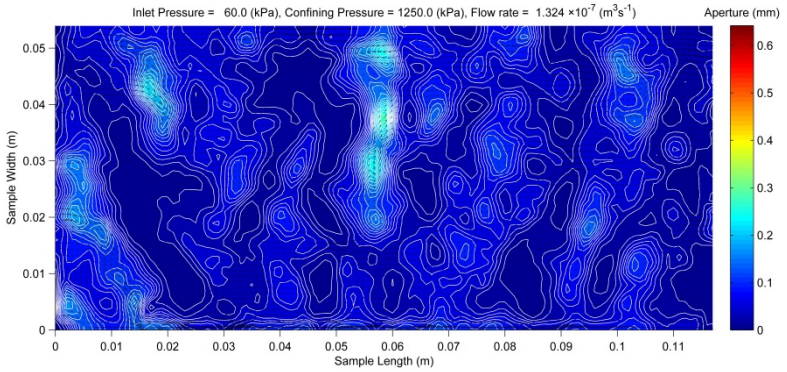
(a)



(b)



(c)



(d)

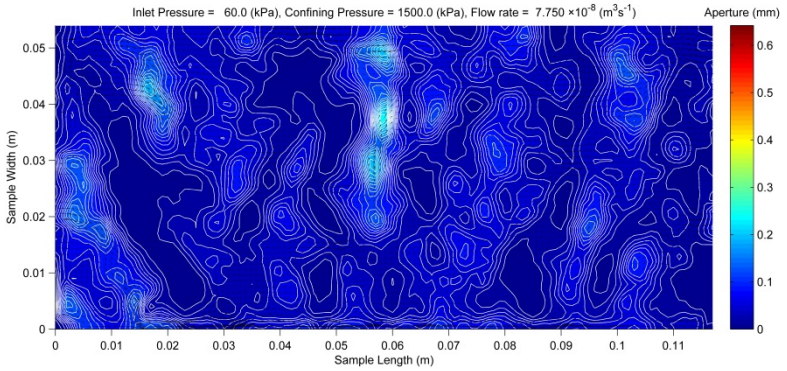


Figure 6.27: Flow simulations of specimen ‘SE’ for increasing confining pressures for 60 kPa inlet pressure. (a) 750 kPa (b) 100 kPa (c) 1250 kPa (d) 1500 kPa

The deformations of aperture when confining pressure is increased were clearly simulated by the programme with the aperture contour plots, and also selection of the flow paths when the fracture aperture contains tight apertures and open apertures were properly simulated by the velocity vector plots and aperture contour plots. The Reynolds number (Re) for rock fracture flows is given by Equation (6.3) where the average aperture ($\bar{e}$) can be used as the characteristic linear dimension (Kishida *et al*. 2013; Singh *et al*. 2014):

$$\mathrm{Re} = \frac{\bar{e}U\rho}{\mu} \tag{6.3}$$

In the above, U is the mean velocity, ρ is the fluid density and μ is the dynamic fluid viscosity. Cross sectional area A of the fracture is equal to $\bar{e}w$ if the fracture width is w. Considering the volumetric flow rate Q is given by UA, Reynolds number was calculated using the Equation (6.5) in this study.

$$\mathrm{Re} = \frac{Q\rho}{\mu w} \tag{6.4}$$

6.5.5 *Simplifications of the Developed Mathematical Model*

The developed mathematical model governing equations (3.39), (3.40), and (3.37) can be further simplified to obtain the cubic formula as follows:

$$\frac{\partial}{\partial t}(\rho e\bar{u}) = -(\rho e\bar{u})\frac{\partial \bar{u}}{\partial \mathrm{x}} - (\rho e\bar{v})\frac{\partial \bar{u}}{\partial \mathrm{y}} + \mu\frac{\partial^2}{\partial \mathrm{x}^2}(e\bar{u}) + \mu\frac{\partial^2}{\partial \mathrm{y}^2}(e\bar{u}) - \frac{12\mu}{e}\bar{u} + \rho \mathrm{g} e\frac{\partial \acute{\mathrm{Z}}}{\partial \mathrm{x}} - e\frac{\partial \mathrm{p}}{\partial \mathrm{x}} \tag{6.5}$$

$$\frac{\partial}{\partial t}(\rho e\bar{v}) = -(\rho e\bar{u})\frac{\partial \bar{v}}{\partial \mathrm{x}} - (\rho e\bar{v})\frac{\partial \bar{v}}{\partial \mathrm{y}} + \mu\frac{\partial^2}{\partial \mathrm{x}^2}(e\bar{v}) + \mu\frac{\partial^2}{\partial \mathrm{y}^2}(e\bar{v}) - \frac{12\mu}{e}\bar{v} + \rho \mathrm{g} e\frac{\partial \acute{\mathrm{Z}}}{\partial \mathrm{y}} - e\frac{\partial \mathrm{p}}{\partial \mathrm{y}} \tag{6.6}$$

$$\frac{\partial(\rho e\bar{u})}{\partial \mathrm{x}} + \frac{\partial(\rho e\bar{v})}{\partial \mathrm{y}} = 0 \tag{6.7}$$

When unidirectional flow is assumed, the $\bar{v}$ component of the velocity vanishes and one-dimensional momentum equation for the velocity can be derived from Equation (3.39) as follows.

$$\frac{\partial}{\partial t}(\rho e\bar{u}) = -(\rho e\bar{u})\frac{\partial \bar{u}}{\partial \mathrm{x}} + \mu\frac{\partial^2}{\partial \mathrm{x}^2}(e\bar{u}) - \frac{12\mu}{e}\bar{u} + \rho \mathrm{g} e\frac{\partial \acute{\mathrm{Z}}}{\partial \mathrm{x}} - e\frac{\partial \mathrm{p}}{\partial \mathrm{x}} \tag{6.8}$$

The continuity Equation (3.37) drops the second term for unidirectional flow, and for steady flow, the transient term on the left of Equation (6.8) vanishes. The first and second terms in RHS of Equation (6.8) are advection and diffusion respectively, and both the terms are dropped to assume a negligible effect to the flow. The remaining terms can then be arranged as follows;

$$\bar{u} = \frac{e^2}{12\mu}\left(\rho \mathrm{g}\frac{\partial \acute{\mathrm{Z}}}{\partial \mathrm{x}} - \frac{\partial \mathrm{p}}{\partial \mathrm{x}}\right) \tag{6.9}$$

For a fracture of width ‘w’ the flow rate can be calculated as given below, which is the cubic formula.

$$Q = \frac{e^3 w}{12\mu}\left(\rho \mathrm{g}\frac{\partial \acute{\mathrm{Z}}}{\partial \mathrm{x}} - \frac{\partial \mathrm{p}}{\partial \mathrm{x}}\right) \tag{6.10}$$

It is therefore clear that the developed model is an extension to the cubic formula. For steady flow, when the diffusion and advection terms are dropped keeping the two-dimensional flow characteristics, Equations (3.39) is reduced to Equation (6.9), and Equation (3.40) becomes (6.11):

$$\bar{v} = \frac{e^2}{12\mu}\left(\rho g \frac{\partial \acute{Z}}{\partial y} - \frac{\partial p}{\partial y}\right) \tag{6.11}$$

Then, substituting Equations (6.9) and (6.11) in Equation (3.37) gives athe Reynolds Equation (6.12).

$$\frac{\partial}{\partial x}\left(\frac{\rho e^3}{12\mu}\left(\rho g \frac{\partial \acute{Z}}{\partial x} - \frac{\partial p}{\partial x}\right)\right) + \frac{\partial}{\partial y}\left(\frac{\rho e^3}{12\mu}\left(\rho g \frac{\partial \acute{Z}}{\partial y} - \frac{\partial p}{\partial y}\right)\right) = 0 \tag{6.12}$$

Koyama (2007) used the Reynolds equation to simulate fracture flow and concluded that the Reynolds equation overpredicts only 5-10 % compared to the Navier-Stokes equation predictions. Reynolds equation can be solved easily and therefore it can be used for fracture flow simulations when the fracture roughness is low (Zimmerman and Yeo (2013).

6.6 Contact Formation Predictions

The treatment of contact as local boundaries was explained in Chapter 4; the module to simulate this was saved separately in the programme because this treatment extends the solution time quite extensively. The predicted flow rates decreased slightly compared to the flow rate calculations where the contacts were replaced with a very small aperture. On this basis the contact formation module can be used for flow simulations while the other can be used to predict the flow rates of rock fractures irrespective of the simulation pattern.

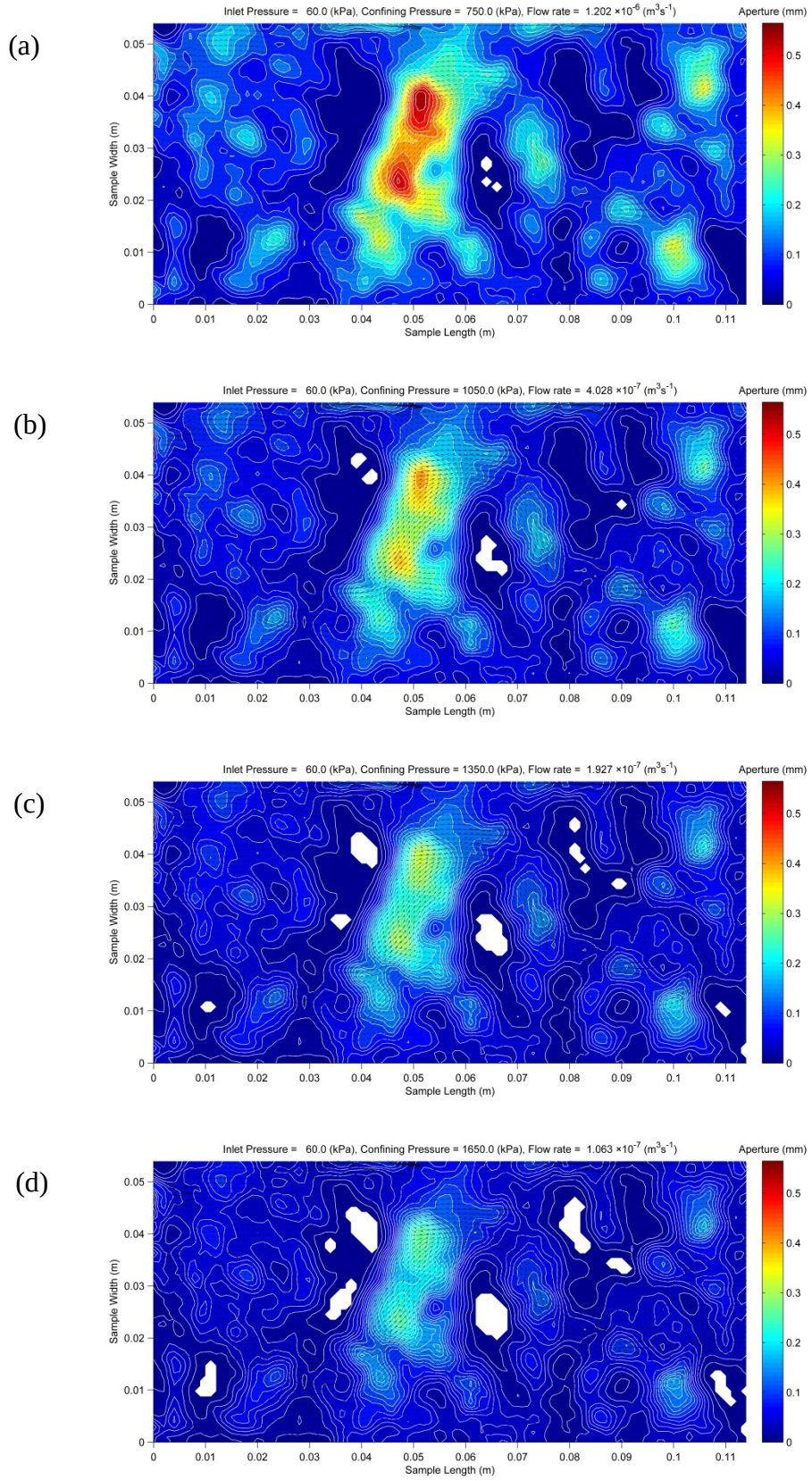


Figure 6.28: Progressive formation of contacts (white patches) for increasing confining pressures for 60 kPa inlet pressure for specimen 'N'. (a) 750 kPa (b) 1050 kPa (c) 1350 kPa (d) 1650 kPa

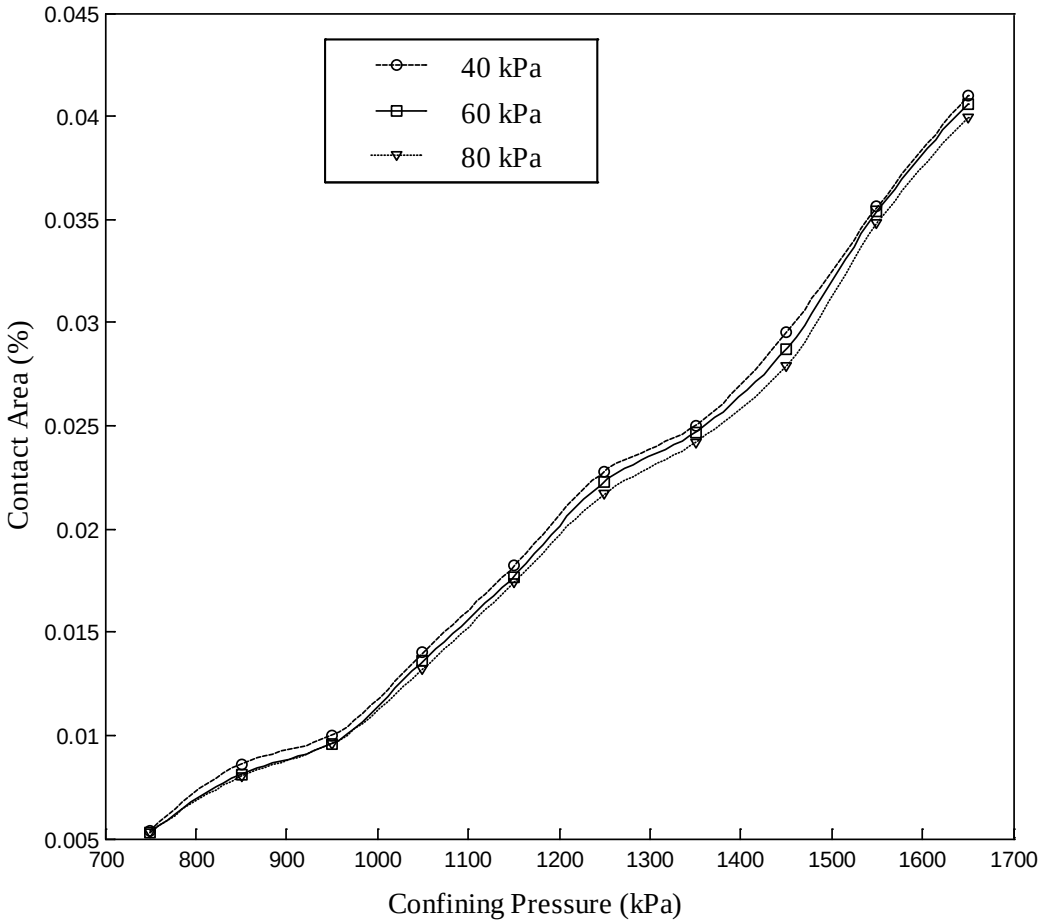


Figure 6.29: Contact area development against the applied confining pressure for 40 kPa, 60 kPa and 80 kPa inlet water pressures

Figure 6.28 shows the contact formation predicted by the programme when the closed apertures were assumed to be local boundaries. The model predictions were almost equal but slightly lower than the predicted flow rates. Figure 6.28 illustrates the formation of contacts in the aperture at increasing confining pressures of 750 kPa, 1050 kPa, 1350 kPa and 1650 kPa at 60 kPa inlet pressure. The zero apertures were plotted in white in the contour plot whereas the formation and growth of contacts in areas with smaller apertures are clearly represented. In this representation no flow vectors that were drawn across the apertures became contacts, which means

the new approach provides a more realistic simulation, because, it treats the contacts as local boundaries for the flow domain. Figure 6.29 shows the increase in the percentage of contact area with the confining pressure increment for three inlet pressures of 40 kPa, 60 kPa, and 80 kPa. According to Figure 6.29, the higher the inlet pressure the lower the percentage of contact area, a result that reflects how the effective stress influenced the deformation.

6.7 Transverse Flow Predictions

The cylindrical specimens used in the triaxial apparatus for fracture permeability cannot be tested for directional permeability because flow can only be measured forwards and backwards in a longitudinal direction, but if the direct shear apparatus is used for flow tests the directional permeability could be measured. However, to use direct shear apparatus for permeability test, the boundaries need to be sealed and only uniaxial stress could be applied, but sealing the boundaries creates problems with the load-deformation behaviour because the seal can be broken and add additional stiffness to the fracture.

According to the computer programme predictions and laboratory results, the programme can predict the real flow behaviour of rock fractures, and therefore it can easily calculate the transverse directional flow in the same sample. Here, the flow domain was flipped by 90^0 to turn the fracture so that one of the long sides became the inlet and the other became the outlet, and then the predictions for increasing confining pressures at different inlet water pressures were calculated the same way. The simulated flow patterns for the sandstone specimens ‘MG’, ‘P’, ‘N’ and ‘SE’ in the transverse direction are shown from Figures 6.30 to 6.33. Only one combination

of inlet pressure and confining pressure for each specimen is shown here and the aperture deformation with increasing confining pressure showed similar behaviour as in longitudinal direction flow.

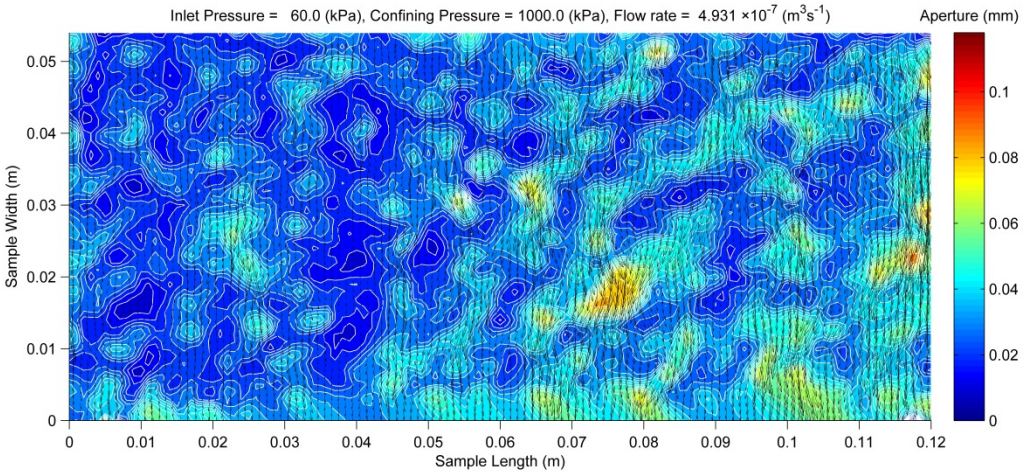


Figure 6.30: Transverse flow simulation for specimen ‘MG’ at 1000 kPa confining pressure and 60 kPa inlet pressure

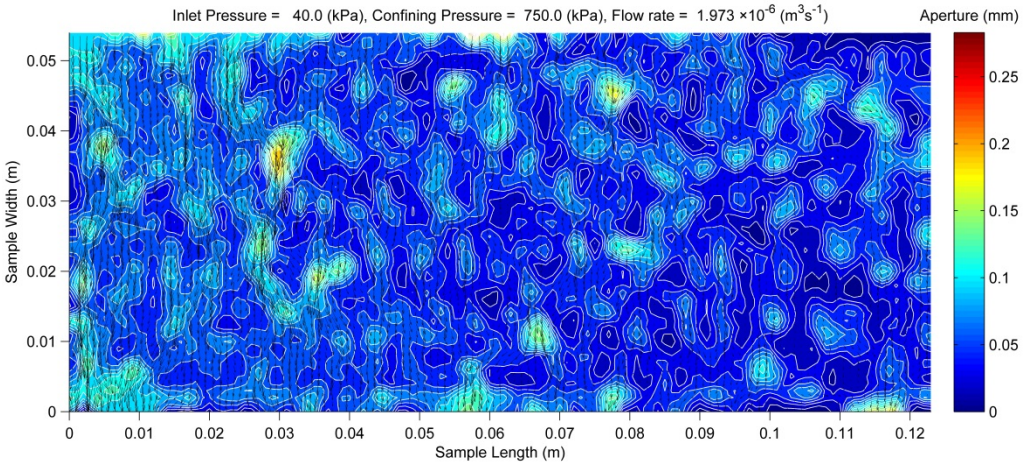


Figure 6.31: Transverse flow simulation for specimen ‘P’ at 750 kPa confining pressure and 40 kPa inlet pressure

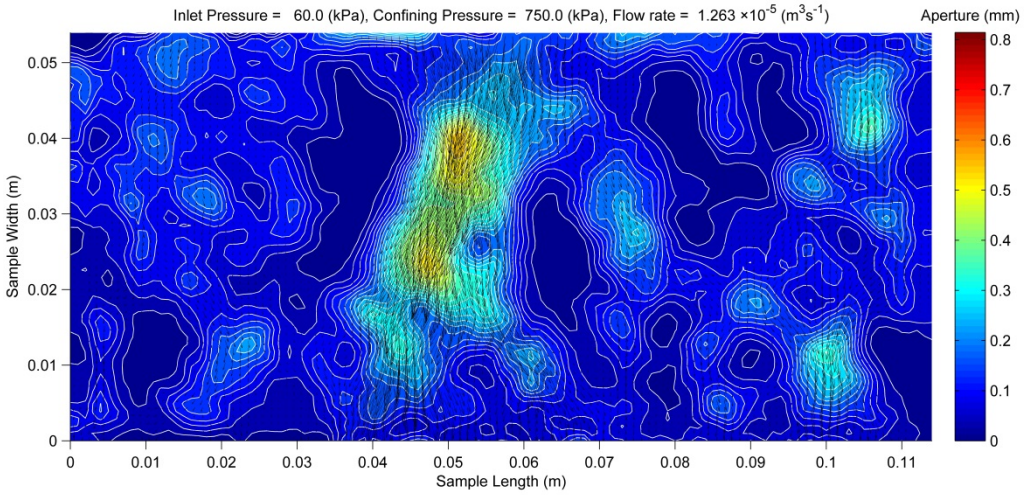


Figure 6.32: Transverse flow simulation for specimen 'N' at 750 kPa confining pressure and 60 kPa inlet pressure

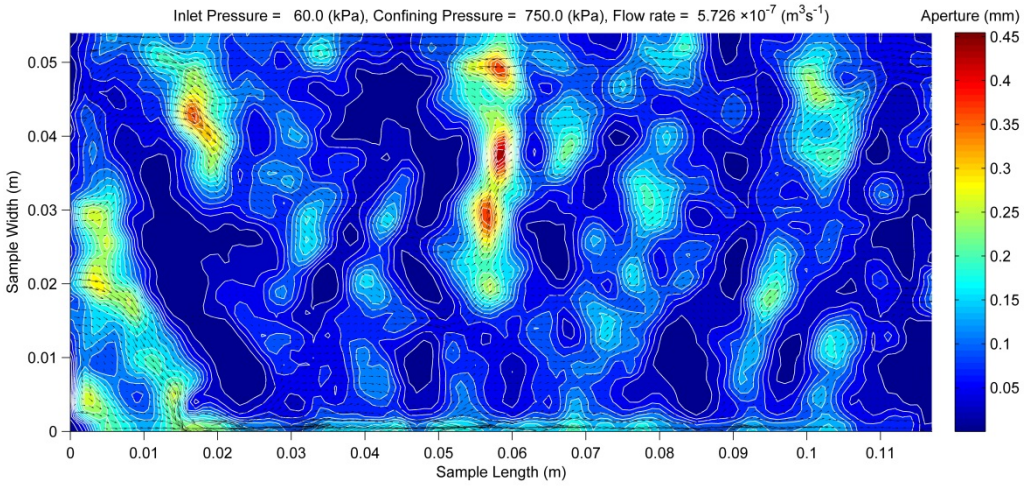


Figure 6.33: Transverse flow simulation for specimen 'SE' at 750 kPa confining pressure and 60 kPa inlet pressure

Behaviour of the flow against confining pressure in a transverse direction was similar to the longitudinal direction flow because fracture deformation is almost the same, irrespective of the direction of flow. To compare the result with longitudinal

directional flow, the macroscopic permeability for the fracture at different confining pressures is calculated in the following section.

6.8 Directional Permeability of Rock Fractures

The directional permeability of rock fractures is important in underground mine design because creating new openings alters the hydraulic gradient. Fracture permeability in one direction can usually be assumed to be equal in forward and backward flows, but flow in an orthogonal direction will be completely different so the permeability of a fracture in an orthogonal direction is needed when flow rates are predicted in mine operations.

From the flow rate simulations shown in the Section 6.7, the macroscopic permeability in the transverse direction at different confining stresses was calculated. The macroscopic permeability in the longitudinal direction was also calculated from the flow simulations carried out in that direction, whereas the macroscopic permeability k of a rock fracture can be calculated from the Equation (6.13).

$$k = \frac{Q\mu}{\nabla P . A} \tag{6.13}$$

Where Q is the volumetric flow rate, μ is the dynamic viscosity of the fluid, ∇P is the macroscopic pressure gradient and A is the cross sectional area of the fracture at the outlet which can be calculated using the following discrete equation:

$$A = \Delta y . \sum_{i=1}^{n} e_{m,i} \tag{6.14}$$

In the above, Δy is the grid size of a finite volume in the direction perpendicular to the direction of flow, m and n are the numbers of discrete volumes in the direction of flow and perpendicular to the flow, respectively.

Tables 6.3 and 6.4 present the directional permeability of the mated and dislocated joint specimens respectively where k_x is the permeability in the longitudinal direction and k_y is the permeability in the transverse direction of the fracture. Both dislocated specimens had open apertures in the transverse direction (Figures 6.11 and 6.12). When water flows in the transverse direction, these apertures would provide an easy path and therefore permeability in that direction is higher, but if water flowed in the longitudinal direction, the small apertures on the path would create a tortuous flow path and reduce permeability in that direction. The mated specimen 'MG' had smaller apertures on one side (Figure 6.9) which made transverse directional flows difficult while the longitudinal flow path did not undulate very much, so the transverse directional permeability was lower than the longitudinal directional permeability. The apertures in specimen 'P' were distributed everywhere in the flow domain (Figure 6.10) that caused the directional permeability to be almost equal in both orthogonal directions.

Table 6.3: Directional permeabilities for the mated fracture specimens at different confining pressures

Confining Pressure (kPa)	MG		P	
	k_x (m^2)	k_y (m^2)	k_x (m^2)	k_y (m^2)
750	2.38×10^{-10}	1.54×10^{-10}	1.76×10^{-10}	1.89×10^{-10}
1000	1.93×10^{-10}	8.75×10^{-11}	1.11×10^{-10}	1.31×10^{-10}
1250	1.25×10^{-10}	5.65×10^{-11}	8.13×10^{-11}	9.61×10^{-11}
1500	9.34×10^{-11}	3.95×10^{-11}	4.19×10^{-11}	7.35×10^{-11}

Table 6.4: Directional permeabilities for the dislocated fracture specimens at different confining pressures

Confining Pressure (kPa)	N		SE	
	k_x (m^2)	k_y (m^2)	k_x (m^2)	k_y (m^2)
750	6.35×10^{-10}	3.64×10^{-09}	7.04×10^{-11}	5.89×10^{-10}
1000	3.59×10^{-10}	2.11×10^{-09}	3.84×10^{-11}	3.41×10^{-10}
1250	2.30×10^{-10}	1.37×10^{-09}	2.45×10^{-11}	2.21×10^{-10}
1500	1.58×10^{-10}	9.64×10^{-10}	1.38×10^{-11}	1.54×10^{-10}

6.9 Summary

This chapter contained the results from the laboratory experiments and computer programme outputs used to verify the proposed model, and a discussion of the fracture flow of rock. Four 54mm diameter specimens of sandstone were tested with an artificially created sub-axial crack to test the fracture permeability. Two of them were kept as mated joints, while the other two were displaced by 3 mm to make a dislocated joint. The specimens were tested for permeability in a high pressure triaxial cell and then the fracture apertures were measured using a 3D scanner.

The experimentally investigated flow behaviours were simulated using the developed MATLAB programme and revealed that the programme can predict the volumetric flow rates of Reynolds numbers below 10. Since underground fracture flows are usually slow, the model can be used to predict underground flow. In fact the model was extended to treat the contact generation upon loading to make the flow simulations more realistic. When the contacts are given very small apertures the

fracture flows can be simulated correctly but the contact still permeates a small amount of fluid. To overcome this problem the apertures that deformed its total spacing were treated as local boundaries with no flow across them. This enabled the contact formation to be simulated properly while the normal load was increased. The programme could also be used to calculate permeability in a transverse direction to the axial direction of the fracture, a result that could not be achieved using the triaxial cell.

Chapter 7 Conclusions and Recommendations

7.1 General

The past literature carried out in rock fracture hydraulics provided a sound platform for this PhD study. Modelling rock fracture flow is an important research area in geotechnical engineering, because, it is often related to underground mining, petroleum extraction, ground water exploration, radioactive waste disposal among other activities. This doctoral thesis presented the development of an equivalent two-dimensional flow model for rough deformable rock fractures together with a numerical analysis and a new computer programme (RFFS: Rock Fracture Flow Solver) to obtain the model solutions, which were verified using laboratory experiments on real jointed rock specimens.

The proposed equivalent two-dimensional flow model was derived from the Navier-Stokes theory by integrating it in the direction of fracture aperture. Fracture aperture of the upper and lower walls was taken as the limits of the integration, and by assuming deformable fractures, the integration limits were treated as functions of space and time. When the N-S equation was integrated over the aperture, a depth averaged velocity was used to depict the velocity as a continuum in the direction of

the aperture. In this process, the velocity distribution over the aperture was assumed to be parabolic with no slip boundary conditions at the fracture walls. One of the main reasons for proposing this new model for real rock fracture flows was to demonstrate the changes in the flow due to aperture changes with the applied stress, and to quantify the flow behaviour with increasing confining pressure. The model predictions and the experimental results both indicated that the relationship between the fracture permeability and applied normal stress can be represented by an appropriate power function.

The finite volume method was used to discretise the model in the flow domain with a structured grid arrangement that had a staggered grid for velocity components. The central differencing scheme was used to approximate the partial derivatives and the equations were solved iteratively. The proposed model was unable to solve directly using the available numerical software, because, the fracture aperture appears in the flow equations as a variable in space. Additionally, the available numerical software require a mesh to be generated to commence a flow simulation. Furthermore, if fracture deformations are considered, the mesh deformation also should be formulated separately. For a quick result on flow simulation, it is beneficial to have a flow simulator which only needs the aperture distribution while no mesh generator is needed. Rock Fracture Flow Solver (RFFS) was developed in MATLAB to solve the proposed model numerically addressing the above criteria to measure flow characteristics through rough walled single rock fracture. The main controlling parameters for RFFS included the initial aperture distribution, the fracture dimensions, the initial joint normal stiffness, the fluid density, the dynamic fluid viscosity, the number of grids needed in the 'X' direction of the fracture, and the

increments of applied confining pressure. In summary, for computed aperture deformation, the program calculated and graphically illustrated the aperture and pressure distributions, plus the velocity vector plots.

Sandstone rock specimens with a sub-axial fracture were tested in the laboratory using the High Pressure Two-Phase Triaxial Apparatus (HPTPTA) for both mated and dislocated fracture specimens. HPTPTA was designed and built at University of Wollongong in early 2000's, but had to be modified by the Writer to conduct the specific tests required for this research work. For the tested fracture specimens, the joint roughness coefficients (JRC) were between 5.5 and 7.5, and they were subjected to confining pressures ranging from 500 to 1750 kPa. Three inlet pressures of 40 kPa, 60 kPa, and 80 kPa were applied for each sample and for each confining pressure. The fracture apertures were measured by replicating the aperture from silicon rubber and then scanning it using a 3D non-contact laser scanner. The specific salient outcomes of this study, as well as limitations and future recommendations, are listed as follows.

7.2 Two-Dimensional Hydro-Mechanical Model for Rough Rock Joints

- The two-dimensional model developed for rock fracture flows can be applied to deformable fractures to predict the flow rates and calculate the correct fracture permeability that was able to overcome errors (overestimation) associated with the conventional cubic flow (Darcy) formula. The proposed model can be reduced to the traditional cubic formula (steady state laminar flow) by omitting the advection and diffusion terms.

- The model was developed by assuming that the fracture walls were deformable, and therefore the strain (deformation) criterion could be coupled with the flow model to predict fluid flows through deforming rock fractures.
- The Bandis well-known hyperbolic deformation model (Equation 3.91) was used in discrete form in the current model, in which each initial aperture was considered as the maximum closure of that particular cell.
- The SIMPLE algorithm was extended to include the aperture deformations subjected to normal loading. The proposed flow model contained the fracture aperture as a variable, and the aperture deformation calculations were incorporated in the algorithm to calculate the pressure and velocity profiles, while the aperture deformed under applied stress.

7.3 Rock Fracture Flow Solver (RFFS)

- The Writer's own computer programme RFFS was presented using the MATLAB programming language. It is user friendly to operate and can customise the fracture flow simulations and associated predictions, when the fracture aperture distribution is known.
- The User does not need to generate a mesh for a fracture flow domain with uneven walls, in order to simulate the fracture flows in this model. This is because, the apertures measured in a preferred resolution can be input as a 2D array, and then the programme will develop a preferred grid size for executing the iterations.

- From the sensitivity analysis performed using a range of grid sizes (4 mm to 0.25 mm), a 1 mm grid size was found to be more than sufficient to reach convergence.
- The programme can calculate flows in both orthogonal directions in a rectangular or square flow domain, which can then be used to predict the directional permeability of a rock fracture.
- A separate module was included to the programme to treat aperture contacts formation while the fracture was being deformed by the applied effective stress. Although the calculation time increased in this module, the formation of contacts was simulated in a more realistic way.

7.4 Rock Fracture Flow Behaviour

- Rock fracture permeability tests were conducted using the High pressure Two-Phase Triaxial Apparatus (HPTPTA) to validate the model. The overall model predictions generally agreed well with the laboratory results, albeit a tolerable margin of error.
- The model predictions for the mated and dislocated joints were acceptable for flows with a Reynolds number less than 10.
- Flow simulations in both orthogonal directions showed that the permeability was higher in the direction where larger apertures were linearly located and unobstructed by contacts. However, the same aperture distribution produced a different permeability when the flow direction changed, because, the location of contacts had more influence on flow than the number of contacts.

7.5 Limitations of the Proposed Model

- Because the rock fracture apertures were very small, the velocity profile in the direction of aperture was limited to a parabolic distribution in this study. However, where the apertures are larger and flow is turbulent, the velocity profile in the direction of the aperture should be modified accordingly.
- In the macroscopic view, the rock fractures are planar features, and indeed the proposed model was derived by assuming the fractures were planar. A discontinuity oriented coordinate system (x,y,z) defined for the fracture assuming the (x,y) plane was located between the two fracture walls, and the axis of the aperture was in the z-direction. However, in microscopic view, the flow had an undulating behaviour in the vertical direction which could not be properly addressed in this model. Additionally, flow through non-planar discontinuities such as folds cannot be modelled using the current model, without making significant modifications.
- The advection term was left out of the numerical solution for simplicity, because, it was less influential for relatively slow flow rates. The non-linearity of the advection term made the numerical approximations impossible and was usually omitted. Therefore, the current numerical solution deviates from accuracy when applied to faster flows that are associated with considerable advection effects.

7.6 Recommendations for Future Studies

- The fracture flow model was extended for multiphase flows and for flows in sediment infilled fractures with erosion. However, this extension was not verified in this PhD study due to the time constraints. The extended model is given in Appendix C, and the relevant modifications to include different phases are expected to be carried out and verified by laboratory experiments in the near future, under a subsequent project.
- Velocity distribution in the direction of aperture should be considered non-parabolic, and then the model should be updated to cater for other velocity profiles in fracture flows.
- Model deformation was considered normal to the fracture only, and the shear behaviour of the fracture should also be considered while integrating the Navier-Stokes equation. This is because it could result in combining the flow model with a shear deformation model to produce an alternative model (coupled shear-flow) for flow through rock fractures.
- The finite volume discretisation used a structured grid where all the control volumes were similar across the entire fracture. The use of a non-structured grid would provide more accurate details in flow simulation where special attention should be given, to flow obstacles (contacts) and flow boundaries.
- The measured directional permeability of fractures in a fracture network should be combined to obtain an equivalent permeability tensor for the rock strata that can be directly used to predict groundwater flows in

underground mines. This can be achieved by using a discrete element model, and it is recommended to adopt UDEC.

References

Amadei, B. and Illangasekare, T. (1992). "Analytical solutions for steady and transient flow in non-homogeneous and anisotropic rock joints." Int. J. rock. Mech. Min. Sci. & Geomech. Abstr. **29**(6): 561-572.

Baker, W. J. (1955). Flow in fissured formations. Proceedings of the Fourth World Petroleum Congress. Rome, Carlo Colombo Publishers. **sect. 11/E:** 379-393.

Bandis, S., Lumsden, A. C. and barton, N. R. (1983). "Fundamentals of rock joint deformation." Rock Mech. Min. Sci. & Geomech. Abstr. **20**: 249-268.

Bart, M. and Shao, J. F. (1998). "Damage mechanics approach for hydromechanical coupling in rock joints under normal stress." Mechanics of jointed and Faulted rock.

Bart, M., Shao, J. F., Lydzba, D. and Haji-Sotoudeh, M. (2004). "Coupled hydromechanical modeling of rock fractures under normal stress." Canadian geotechnical journal **41**(4): 686-697.

Batchelor, G. K. (1967). An introduction to fluid dynamics. New York, Cambridge University Press.

Bear, J., Tsang, C. F. and de Marsily, G. (1993). Flow and contaminant transport in fractured rock, Academic Press.

Brown, S. R. (1987). "Fluid flow through rock joints: The effect of surface roughness." Geophysical research **92**(B2).

Brown, S. R. and Scholz, C. H. (1986). "closure of rock joints." Journal of Geophysical Research: Solid Earth **91**(B5): 4939-4948.

Carroll, M. M. (1979). "Effective Stress Law for Aniosotropic Elastic Deformation" Journal of Geophysical Research **84**(B13): 7510-7512.

de Oliveira, C. (1999). GEM: A Finite Element Mesh Generator. Internal Report. London, Imperial Cllege.

Dhowian, A. W. (1980). "The effect of pore pressure on the deformability of jointed rock specimens." Engineering science **6**(1).

Farmer, I. W. (1983). Engineering behaviour of rocks. London, Chapman & Hall.

Gale, J. E. (1977). A numerical field and laboratory stidy of flow in rocks with deformable fractures.

Gangi, A. F. (1978). "Variation of whole and Fractured Porous Rock Permeability with Confining Pressure." International Journal of Rock Mechanics and Mining Science & Geomechanics Abstracts **15**: 249-257.

Garg, S. K. and Nur, A. (1973). "Effective stress laws for fluid-saturated porous rocks." Geophysical research **78**(26).

Garitte, B., Gens, A., Vaunat, J. and Armand, G. (2014). "Thermal Conductivity of Argillaceous Rocks: Determination Methodology Using In Situ Heating Tests." Rock Mechanics and Rock Engineering **47**(1): 111-129.

Ge, S. (1997). "A governing equation for fluid flow in rough fractures." Water Resour. Res. **33**(1): 53-61.

Gens, A., Carol, I. and Alonso, E. E. (1990). "A constitutive model for rock joints formulation and numerical implementation." Computers and Geotechnics **9**(1–2): 3-20.

Gens, A., Garitte, B. and Vaunat, J. (2008). Modelling the Response of Argillaceous Rocks in Underground Excavations. 12th International Conference of IACMAG. Goa, India**:** 1197-1209.

Goodman, R. E. (1974). Mechanical properties of joints. 3rd Congress of ISRM, Denver.

Goodman, R. E. (1976). Methods of Geological Engineering in Discontinuous Rock. New York, West.

Harlow, F. H. and Welch, J. E. (1965). "Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface." Physics of Fluids **8**: 2182-2189.

Indraratna, B. and Haque, A. (1999). "Triaxial equipment for measuring the permeability and strength of intact and fractured rocks." Geotechnique **49**(4): 515-521.

Indraratna, B., Haque, A. and Gale, W. (1998). Evaluation of jointed rock permeability using a high pressure triaxial apparatus. Mechanics of Jointed and Faulted Rock, Balkema**:** 561-566.

Indraratna, B. and Oliveira, D. A. F. (2010). "A shear-displacement criterion for soil-infileed rock discontinuities." Geotechnique **60**(8).

Indraratna, B., Price, J. and Gale, J. (2002). Fourier description of fracture roughness. 5th North American Rock Mechs Symp., University of Torronto Press.

Indraratna, B., Price, J., Ranjith, P. and Gale, W. (2002). "Some aspects of unsaturated flow in jointed rock." International Journal of Rock Mechanics and Mining Sciences **39**(5): 555-568.

Indraratna, B. and Ranjith, P. (2001). Hydromechanical Aspects and Unsaturated Flow in Jointed Rock, A. A. Balkema Publishers.

Indraratna, B., Ranjith, P., Price, J. and Gale, W. (2003). "Two-Phase (Air and Water) Flow through Rock Joints: Analytical and Experimental Study." Geotechnical and Geoenvironmental Engineering **129**(10): 918-928.

Indraratna, B. and Ranjith, P. G. (2001). "Laboratory mesurement of two-phase flow parameters on high pressure triaxial testing." Journal of Geotechnical and Geoenvironmental Engineering **127**(6): 530-542.

Indraratna, B., Ranjith, P. G. and Gale, W. (1999). "Single phase water flow through rock fractures." Geotechnical & Geological Engineering **17**(3-4): 211-240.

Issa, R. I. (1986). "Solution of the implicitly discretised fluid flow equations by operator-splitting." J. Comput. Phys. **62**(1): 40-65.

Iwai, L. (1976). Fundamental studies of fluid flow through a single fracture. Berkerley, university of California. **PhD:** 208.

Jones, F. O. (1975). "A laboratory study of the effects of confining pressure on fracture flow and storage capacity in carbonate rocks : 8F, 22R. J. Pwtroleum Tech. V27, JAN. 1975, P21-27." International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts **12**(4): 55-55.

Khalili-Naghadeh, N. and Valliappan, S. (1991). "Flow through fissured porous media with deformable matrix: Implicit formulation." Water resources research **27**(7): 1703-1709.

Khalili, N., Habte, M. A. and Zargarbashi, S. (2008). "A fully coupled flow deformation model for cyclic analysis of unsaturated soils including hydraulic and mechanical hystereses." Computers and Geotechnics **35**(6): 872-889.

Khalili, N. and Selvadurai, A. P. S. (2003). "A fully coupled constitutive model for thermo-hydro-mechanical analysis in elastic media with double porosity." Geophysical Research Letters **30**(24): 2268.

Khalili, N. and Valliappan, S. (1996). Unified theory of flow and deformantion in double porous media. Amsterdam, PAYS-BAS, Elsevier.

Kishida, K., Sawada, A., Yasuhara, H. and Hosoda, T. (2013). "Estimation of fracture flow considering the inhomogeneous structure of single rock fractures." Soils and Foundations **53**(1): 105-116.

Koyama, T. (2007). Stress, Flow and Particle Transport in Rock Fractures. Department of Land and Water Resources Engineering. Stockholm, Royal Institute of Technology (KTH),. **PhD:** 289.

Koyama, T., Li, B., Jiang, Y. and Jing, L. (2009). "Numerical modelling of fluid flow tests in a rock fracture with a special algorithm for contact areas." Computers and Geotechnics **36**(1): 291-303.

Koyama, T., Neretnieks, I. and Jing, L. (2008). "A numerical study on differences in using Navier–Stokes and Reynolds equations for modeling the fluid flow and particle transport in single rock fractures with shear." International Journal of Rock Mechanics and Mining Sciences **45**(7): 1082-1101.

Kranz, R. L., Frankel, A. D., Engelder, T. and Scholz, C. H. (1979). "The permeability of whole and jointed Barre granite." rock Mech. Min. Sci. & Geomech. Abstr. **16**: 225-234.

Kulhawy, F. H. (1975). "Stress deformation properties of rock and rock discontinuities." Engineering Geology **9**(4): 327-350.

Lee, H. S. and Cho, T. F. (2002). "Hydraulic Characteristics of Rough Fractures in Linear Flow under Normal and Shear Load." Rock Mechanics and Rock Engineering **35**(4): 299-318.

Li, B., Jiang, Y., Koyama, T., Jing, L. and Tanabashi, Y. (2008). "Experimental study of the hydro-mechanical behavior of rock joints using a parallel-plate model containing contact areas and artificial fractures." International Journal of Rock Mechanics and Mining Sciences **45**(3): 362-375.

Lomize, G. M. (1951). Flow in Fractured Rocks (English Translation). Gosenergoizdat, State Press

Neuzil, C. E. and Tracy, J. V. (1981). "Flow through fractures." Water Resource Res. **17**(1): 191-199.

Nguyen, T. S. and Selvadurai, A. P. S. (1998). "A model for coupled mechanical and hydraulic behaviour of a rock joint." International Journal for Numerical and Analytical Methods in Geomechanics **22**(1): 29-48.

Nur, A. and Byerlee, J. D. (1971). "An exact effetive stress law for elastic deformation of rock with fluids." Geophysical research **76**(26).

Oliveira, D. A. F. and Indraratna, B. (2010). "Comparison between models of rock discontinuity strength and deformation." Journal of Geotechnical and Geoenvironmental Engineering **136**(6): 864-874.

Pak, A. and Chan, D. H. (2004). "A Fully Implicit Single Phase T-H-M Fracture Model for Modelling Hydraulic Fracturing in Oil Sands."

Patanker, S. V. (1980). Nemerical heat transfer and fluid flow. New York, McGraw Hill Book Company.

Patanker, S. V. and Spalding, D. B. (1972). "A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows." Journal of Heat and Mass Transfer **15**(10): 1787-1806.

Patir, N. and Cheng, H. S. (1978). "An average flow model for determining effects of three-dimensional roughness on partial hydrodynamic lubrication." Journal of Lubrication Technology **100**: 12-17.

Price, J. and Indraratna, B. (2005). "Development of an equivalent homogenous fluid model for pseudo-two-phase (air + water) flow through fractured rock." Journal of Geotechnical and Geoenvironmental Engineering **131**(7): 857-866.

Pruess, K. and Tsang, Y. W. (1990). "On two phase relative permeability and capillary pressure of rough walled rock fractures." Water Resource Res. **26**(9): 1915-1926.

Ranjith, P. G. and Darlington, W. (2007). "Nonlinear single-phase flow in real rock joints." Water Resour. Res. **43**(9): W09502.

Raven, K. G. and Gale, J. (1985). "Water Flow in a Natural Rock Fracture as a Function of Stress and Sample Size." Rock Mech. Min. Sci. & Geomech. Abstr. **22**(4): 251-261.

Schrauf, T. W. and Evans, D. D. (1986). "Laboratory Studies of Gas Flow Through a Single Natural Fracture." Water resources research **22**(7): 1038-1050.

Snow, D. T. (1968). "Rock fracture spacings, openingsand porosities." Soil Mech. Found. Div. ASCE **94(SMI)**: 73-91.

Tao, Q. and Ghassemi, A. (2009). Investigation of stress-dependent fracture permeability in naturally fractured reservoirs using a fully coupled poroelastic displacement discontinuity model. SPE Annual Technical Conference and Exhibition. Orleans.

Tsang, Y. W. (1984). "The Effect of Tortuosity on Fluid Flow Through a Single Fracture." Water Resource Res. **20**(9): 1209-1215.

Tsang, Y. W. and Witherspoon, P. A. (1981). "hydromechanical behaviour of a deformable rock fracture subject to normal stress." Geophysical research **86**(B10).

Tse, R. and Cruden, D. M. (1979). "Estimating joint roughness coefficients." International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts **16**(5): 303-307.

Van Doormaal, J. P. and Raithby, G. D. (1984). "Enhancements of the SIMPLE method for predicting incompressible fluid flows." Numerical Heat Transfer **7**(2): 147-163.

Walsh, J. B. (1981). "Effect of Pore Pressure and Confining Pressure on Fracture Permeability." International Journal of Rock Mechanics Mineral Science and Geomechanics Abstracts **18**: 429-435.

Witherspoon, P. A., Wang, J. S. Y., Iwai, K. and Gale, J. E. (1980). "Validity of Cubic Law for Fluid Flow in a Deformable Rock Fracture." Water Resource Res. **16**(6): 1016-1024.

Xiao, B., Carter, J. P. and Alehossein, H. (1994). "Elastoplastic analysis of jointed rocks using a coupled finite element and boundary element method." International Journal for Numerical and Analytical Methods in Geomechanics **18**(7): 445-466.

Yoshinaka, R. and Yamabe, T. (1986). "Joint stiffness and the deformation behaviour of discontinous rock." Rock mechanics and mineral science **23**(1).

Zhang, Z. (2013). Hydromechanical behaviour and nonlinear flow characteristics of rock fractures. School of Civil, Mining and Environmental Engineering. Wollongong, University of Wollongong. **PhD**.

Zhang, Z. and Nemcik, J. (2013). "Friction Factor of Water Flow Through Rough Rock Fractures." Rock Mechanics and Rock Engineering **46**(5): 1125-1134.

Zhou, C. B., Sharma, R. S., Chen, Y. F. and Rong, G. (2008). "Flow-stress coupled permeability tensor for fractured rock masses." Numerical and analytical methods in geomechanics **32**.

Zimmerman, R. W., Al-Yaarubi, A., Pain, C. C. and Grattoni, C. A. (2004). "Non-linear regimes of fluid flow in rock fractures." International Journal of Rock Mechanics and Mining Sciences **41**: 163-169.

Zimmerman, R. W. and Bodvarsson, G. S. (1996). "Hydraulic conductivity of rock fractures." Transport in Porous Media **23**(1): 1-30.

Zimmerman, R. W., Chen, D.-W. and Cook, N. G. W. (1992). "The effect of contact area on the permeability of fractures." Journal of Hydrology **139**(1–4): 79-96.

Zimmerman, R. W. and Yeo, I.-W. (2013). Fluid Flow in Rock Fractures: From the Navier-Stokes Equations to the Cubic Law. Dynamics of Fluids in Fractured Rock, American Geophysical Union**:** 213-224.

Appendix A: MATLAB Programme Codes

A.1 Components of a MATLAB Function

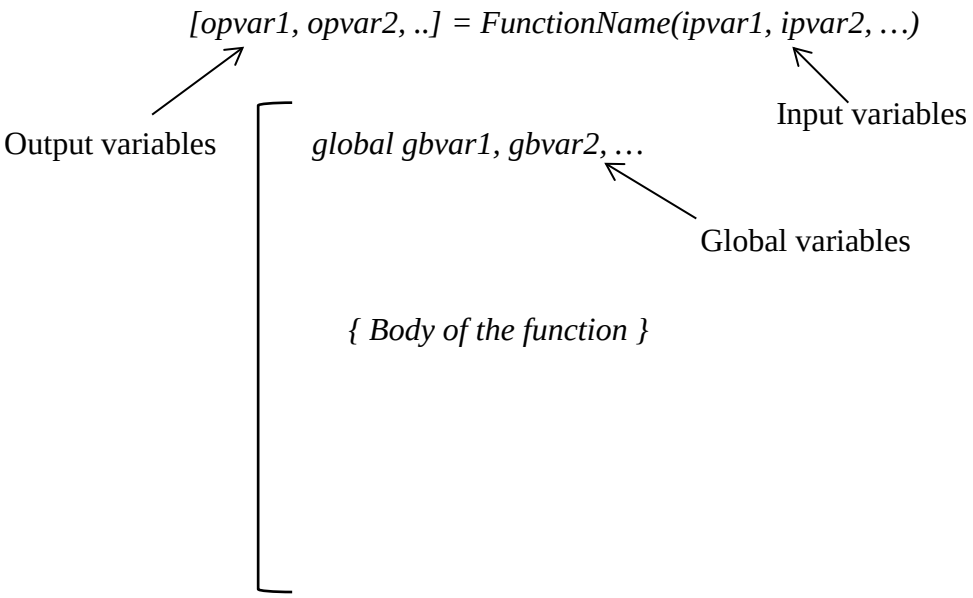


Figure A.1: Components of a typical MATLAB function

Figure A.1 shows the basic components of a Matlab function. When the function is called, the required input parameters should be given. Then the calculation is taken place in the body of the function using the input variables. If global variables are used, they should be declared at the beginning of the function. Values for the output variables should be assigned in the body while calculations are performed. At the end of the function or when the command ‘return’ appears, the

output variables are returned to the place where the function was called. The required number of variables should be kept at the calling of function to catch the returned output variables. Texts followed by % mark are considered as comments and omitted when executing the computation. Capital and simple form of a letter are treated as two characters in Matlab.

A.1.1 Loops used in the functions

For the calculations repetitively, MATLAB provides two basic loops namely FOR loop and WHILE loop. FOR loop is used when repetitive calculations are performed based on continuously increasing or decreasing variable and the number of repetitions are known. On the other hand, WHILE loop can be used for the cases of repetitions are unknown and loop can be run until some condition is satisfied. Early exits from the loops also possible by using the command 'break' at any point inside the loop. And also the command 'continue' is used to skip the rest of the loop and go to the next cycle. The formats of these two loops are illustrated in the Figure 1 and 2 below and these loops are used in the following codes of the functions.

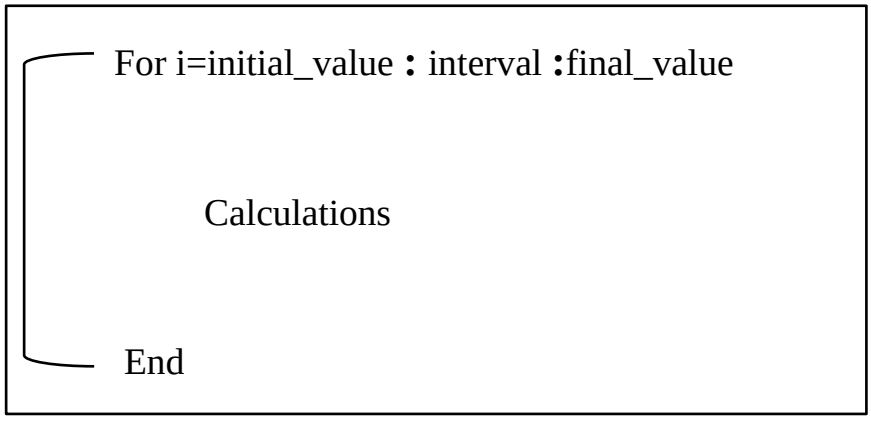


Figure 1: FOR loop structure

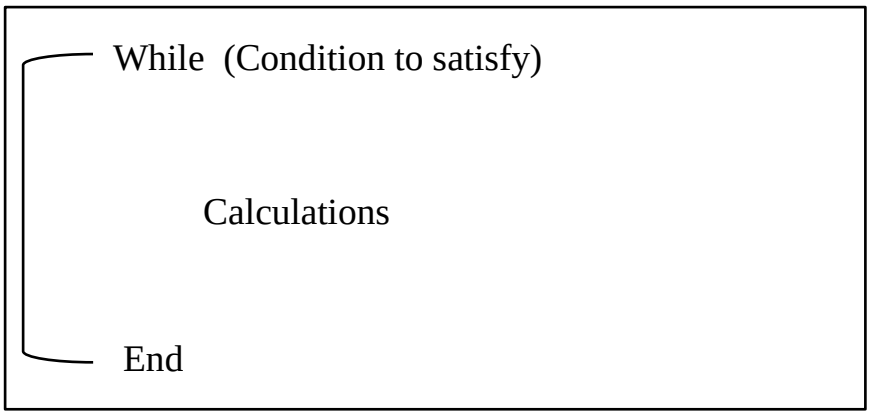


Figure 2: WHILE loop structure

A.2 Opening GUI

The opening graphical user interface is controlled by the *main( )* function. This function is generated by the Matlab itself when the GUI is designed and the main function should not be edited. Additional operations to be done while the GUI is being loaded can be added to the *main_OpeningFcn( ).* The variables that have to be used in more than one function are declared as global variables. In the opening function, the variables to save the user preferences are declared and the images are loaded.

A.2.1 *main_OpeningFcn( )*

```
function main_OpeningFcn(hObject, eventdata, handles, varargin)

global pp mp tm
pp=1;mp=1;tm=1;
handles.output = hObject;

set(0,'CurrentFigure',hObject);
set(hObject,'CurrentAxes',handles.axes2);
dom=imread('domn.jpg');
imshow(dom);

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes main wait for user response (see UIRESUME)
```

```
% uiwait(handles.main);
set(0,'Units','Pixels');
scn=get(0,'ScreenSize');
set(hObject,'Units','Pixels');
fs=get(hObject,'OuterPosition');
mpo=[10, scn(4)-fs(4),fs(3),fs(4)];
set(hObject,'OuterPosition',mpo);
```

Then for the solve command button and aperture load button, a call-back function is created (*solve_Callback( )*) and for the selection radio buttons, separate selection change functions are created (eg: *uipanel13_SelectionChangeFcn( )*).

A.2.2 *solve_Callback( )*

```
function solve_Callback(hObject, eventdata, handles)

global mp tm
texts=findobj('Style','edit');
[n,~]=size(texts);
for i=1:n
    tt=get(texts(i),'String');
    if isempty(tt)
        msgbox('Some fields are empty')
        return
    end
end
h=findobj('type','figure','Tag','vplot');
if isempty(h)==0
    close(h);
end
if tm==1
    if mp==1
        ImplicitSIMPLE(handles)
    elseif mp==2
        SIMPLEwithCONTACTS(handles)
    end
elseif tm==2
    if mp==1
        TransientSIMPLECwithExplicitAdvection(handles)
    elseif mp==2
        TransientSIMPLEwithExplicitAdvection(handles)
    end
end
```

A.2.3 *LoadApertureMatrix_Callback( )*

```
function LoadApertureMatrix_Callback(hObject, eventdata, handles)

[afile,apath,~]=uigetfile({'*.dat','DAT-files (*.mat)';'*.mat','MAT-
files (*.mat)';},'Select aperture data file','e.dat');
set(handles.edit33,'String',strcat(apath,afile));
```

A.2.4 *Sample radio button selection control function*

```
function uipanel10_SelectionChangeFcn(hObject, eventdata, handles)

global mp
switch get(eventdata.NewValue,'Tag') % Get Tag of selected object.
    case 'si'
        mp=1;
    case 'sl'
        mp=2;
end
```

Here uipanel10 is the name of the frame that all the option radio buttons are grouped. A global variable is declared to save the user preference and then that variable is used in the SIMPLE algorithm. These types of functions are created for selecting solution method, flow type, contour plot parameter selection, and flow direction selection too.

A.3 SIMPLE Algorithm

The function to execute the SIMPLE algorithm with treatment to contacts as local boundaries to solve the mathematical flow model and its supporting functions are given below.

A.3.1 *SIMPLEwithcontacts(handles)*

```
function SIMPLEwithCONTACTS(handles)
    global lx ly hx hy nx ny den meu inletp outletp conf ...
           x y  maxcy cnfs flo ape pp epath accu U V confini ...
           SE SP INP g prelax vrelax nthb sthb ki tetax tetay ...
           SU  SV SUE direc
%% Initialization
    getcoeff(handles) % copy user inputs
    w=exist(epath,'file');
    if w==0
        msgbox('Error reading file' ,'Error');
        return
    end
    inie=load(epath);
    if direc==2
        [onx,ony]=size(inie);
        inie=rot90(inie,1);
```

```
        tm=lx;lx=ly;ly=tm;
        nx=round(nx*ony/onx);
    end
    [e, hx, hy, ny]=descret(inie,lx,ly,nx);
    INP.inaper=e;
    emaxc=e*1.001;
    mie=1e-8;
    zer=0;
    [eu, ev]=center2uv(e);
    x = linspace(0,lx,nx+1);
    y = linspace(0,ly,ny+1);
% Variable initialization
    P=zeros(nx,ny);
    Pxi=(outletp-inletp)/lx;
%initial pressure guess assuming uniform pressure gradient
    for kk=1:nx;
       P(kk,:)=inletp+Pxi*(2*kk-1)*lx/(2*nx);
    end
    U=zeros(nx+1,ny);%-eu.^2*Pxi/12/meu*0.1;
    V = zeros(nx+2,ny-1);
%% Initial boundary conditions
    Pi=zeros(ny,1)'+inletp;
    Po=zeros(ny,1)'+outletp;
    zx=sin(tetax);
    zy=sin(tetay);
% Plotting figure generation
    vvh=hgload('vvectorplot.fig');
    set(0,'Units','Pixels');
    sc=get(0,'ScreenSize');
    mh=findobj('Type','Figure','Tag','main');
    set(vvh,'Units','Pixels');
    set(mh,'Units','Pixels');
    mp=get(mh,'OuterPosition');
    vp=get(vvh,'OuterPosition');
    np=[mp(1)+mp(3)+5,sc(4)-vp(4)-25,vp(3),vp(4)];
    set(vvh,'OuterPosition',np);
% Initial deformation
    dntp=e*0;
    effsigman=conf-P;
    dnt=(effsigman.*emaxc)./(effsigman+ki*emaxc);
    dntc=dnt-dntp;
    dntp=dnt;
    e=e-dntc;
    e(e<mie)=zer;
% Calculation commences...
%% while loop for confining pressure changing
    cnc=1;
    cnfs=0;flo=0;ape=0;
    while cnc~=0    %cnc counts number of conf. stress increments
% while loop for solution of one confining pressure
        counter=1;
        nancheck=0;
        Uold=0;
        Vold=0;
        while counter<maxcy
%
%% Solve momentum equation for intermediate velocities
% eq (5.54), (5.55)
            vcou=0;
```

```
            [eu, ev]=center2uv(e);
            eub1=[2*eu(1,:)-eu(2,:);eu;2*eu(end,:)-eu(end-1,:)];
            eub=[2*eub1(:,1)-eub1(:,2) eub1 ...
                2*eub1(:,end)-eub1(:,end-1)];
            evb1=[2*ev(1,:)-ev(2,:);ev;2*ev(end,:)-ev(end-1,:)];
            evb=[2*evb1(:,1)-evb1(:,2) evb1 ...
                2*evb1(:,end)-evb1(:,end-1)];
            minape=min(min(e));
            eub(eub<minape)=minape;
            evb(evb<minape)=minape;
            while vcou<maxcy
% make local B velocities zero
            [uLid, uRid, vDid, vUid]=localb(e);
            U(uLid)=0;U(uRid)=0;
            V(vDid)=0;V(vUid)=0;
      % boundary matrices
                if nthb==1
                    Un=[0;U(:,end)*0;0];    %0
                    Vn=V(:,end)*0;    %0
                elseif nthb==2
                    Un=[U(1,end);U(:,end);U(end,end)];
                    Vn=V(:,end);
                end
                if sthb==1
                    Us=[0;U(:,1)*0;0];      %0
                    Vs=V(:,1)*0;      %0
                elseif sthb==2
                    Us=[U(1,1);U(:,1);U(1,end)];
                    Vs=V(:,1);
                end
                UB1=[U(1,:);U;U(end,:)];
                UB=[Un UB1 Us];
                VB1=[V(1,:);V;V(end,:)];
                VB=[VB1(:,1)*0 VB1 VB1(:,end)*0];
                % local B pressures
                P(e==0)=0;
                PB1=[Pi;P;Po];
                DPX=[(PB1(2,:)-PB1(1,:))/(hx/2);diff(P)/hx; ...
                    (PB1(end,:)-PB1(end-1,:))/(hx/2)];
                DPY=diff(PB1')'/hy;
                % make local B pressure gradients zero
                DPX(uLid)=0;DPX(uRid)=0;
                DPY(vDid)=0;DPY(vUid)=0;


      % neighbour U
                UiMj=UB(1:end-2,2:end-1);
                UiPj=UB(3:end,2:end-1);
                UijM=UB(2:end-1,1:end-2);
                UiJP=UB(2:end-1,3:end);
      % neighbour e at U
                euim=eub(1:end-2,2:end-1);
                euip=eub(3:end,2:end-1);
                euJm=eub(2:end-1,1:end-2);
                euJp=eub(2:end-1,3:end);
      % neighbour V
                VIMj=VB(1:end-2,2:end-1);
                VIPj=VB(3:end,2:end-1);
```

```
            VIjM=VB(2:end-1,1:end-2);
            VIjP=VB(2:end-1,3:end);
        % neighbour e at V
            evIm=evb(1:end-2,2:end-1);
            evIp=evb(3:end,2:end-1);
            evjm=evb(2:end-1,1:end-2);
            evjp=evb(2:end-1,3:end);
        % RHS
            BiJ=-12*meu./eu-2*meu*eu/hx^2-2*meu*eu/hy^2;
            BIj=-12*meu./ev-2*meu*ev/hx^2-2*meu*ev/hy^2;
        %
            UNEW=-(+(den*g*eu*zx)-eu.*DPX)./BiJ ...
                  -((meu*euim/hx^2).*UiMj)./BiJ...
                  -((meu*euip/hx^2).*UiPj)./BiJ...
                  -((meu*euJm/hy^2).*UijM)./BiJ...
                  -((meu*euJp/hy^2).*UiJP)./BiJ;
        %
            VNEW=-(+(den*g*ev*zy)-(ev.*DPY))./BIj...
                 -((meu*evIm/hx^2).*VIMj)./BIj...
                 -((meu*evIp/hx^2).*VIPj)./BIj...
                 -((meu*evjm/hy^2).*VIjM)./BIj...
                 -((meu*evjp/hy^2).*VIjP)./BIj;

             % correct local B velocities in UNEW and VNEW
             UNEW(uLid)=0;UNEW(uRid)=0;
             VNEW(vDid)=0;VNEW(vUid)=0;
            vcou=vcou+1;
            if  max(max(abs((U*(1-vrelax)+UNEW*vrelax-U)./ ...
              (U+eps))))<accu && max(max(abs((V*(1-vrelax) ...
               +VNEW*vrelax-V)./(V+eps))))<accu
                break
            end
            if any(isnan(U))
                nancheck=1;
                break
            end
            if vcou==(maxcy-1)
                uerr=num2str(max(max(abs((U-U*(1-vrelax) ...
                    +UNEW*vrelax)./U))));
                verr=num2str(max(max(abs((V-V*(1-vrelax) ...
                    +VNEW*vrelax)./V))));
                qstr={'Maximum cycles reached in velocity
calculation. Do you want to increase maximum cycles?';'';strcat('
Maximum errors: U = ',uerr,', V = ',verr)};
                ann=questdlg(qstr,'Edit','Yes','No','Yes');
                if strcmp(ann,'Yes')
                    addi=inputdlg('Enter number of cycles to be
added');
                    if isempty(addi)
                        return
                    end
                    maxcy=maxcy+str2num(cell2mat(addi));
                end
            end
            U=U*(1-vrelax)+UNEW*vrelax;
            V=V*(1-vrelax)+VNEW*vrelax;
        end
        if nancheck==1
```

```
                msgbox({'Divergence occured, calculation will stop
now.';'';'Select number of nodes so that a cell is close to a
square'},'Information');
                vh=findobj('Type','Figure','Tag','vplot');
                close(vh)
                return
            end
%% Solve pressure correction. Equation (5.68)
            pcou=0;
            PC=P*0;
% coefficient matrices
                BiJp=-12*meu./eu-2-meu*eu/hx^2-meu*eu/hy^2;
                BIjp=-12*meu./ev-2-meu*ev/hx^2-meu*ev/hy^2;
                BIjBp=[BIjp(:,1) BIjp BIjp(:,end)];
                MiJ=-(den*eu(2:end,:).^2)./(BiJp(2:end,:)*hx^2);
                MimJ=-(den*eu(1:end-1,:).^2)./(BiJp(1:end-
1,:)*hx^2);
                MIj=-(den*evb(3:end-2,2:end).^2)./(BIjBp(2:end-
1,2:end)*hy^2);
                MIjm=-(den*evb(3:end-2,1:end-1).^2)./(BIjBp(2:end-
1,1:end-1)*hy^2);
                MIJ=-MiJ-MimJ-MIj-MIjm;
% RHS
                UX=den*e.*(diff(U)/hx);
                VY=den*e.*(diff(([Vn(2:end-1) V(2:end-1,:) Vs(2:end-
1)])')'/hy);

            while pcou<maxcy
        % boundary matrix
                PCB1=[Pi*0;PC;Po*0];
                PCB=[PCB1(:,1) PCB1 PCB1(:,end)]; %dp/dn=0 at wall
                [PIpJ,PImJ,PIJp,PIJm]=pneibr(PCB);
        %
                PCNEW=-(MiJ.*PIpJ)./MIJ-(MimJ.*PImJ)./MIJ...
                      -(MIj.*PIJp)./MIJ-(MIjm.*PIJm)./MIJ...
                      +UX./MIJ+VY./MIJ;
                PCNEW(e==0)=0;
                pcou=pcou+1;
                if all(abs((PC*(1-prelax)+PCNEW*prelax-
PC)./(PC+eps))<accu)
                    break
                end
                PC=PC*(1-prelax)+PCNEW*prelax;
            end
        %Pressure correction
                PC=PC*prelax;
                P=P+PC;
        %Velocity correction
                Pcu=[((PC(1,:)-Pi*0)/hx);diff(PC(1:end,:))/hx;(Po*0-
PC(end,:))/hx];
                Pcu(uLid)=0;Pcu(uRid)=0;

Pcv=[diff(PC(1,:)/2)/hy;diff(PC')'/hy;diff(PC(end,:)/2)/hy];
                Pcv(vDid)=0;Pcv(vUid)=0;
                Uc=eu.*Pcu./BiJ;
                Vc=ev.*Pcv./BIj;
                U=prelax*Uc+U;
                V=prelax*Vc+V;
                %
```

```
                set(handles.text33,'String',num2str(counter));
                %
                if counter==(maxcy-1)
                     uerr=num2str(max(max(abs((U-Uold)./U))));
                    verr=num2str(max(max(abs((V-Vold)./V))));
                    perr=num2str(max(max(abs(PC./P))));
                    qstr={'Maximum cycles reached. Do you want to
increase maximum cycles?';'';strcat(' Maximum errors: U = ',uerr,',
V = ',verr,', P = ',perr)};
                    ann=questdlg(qstr,'Edit','Yes','No','Yes');
                    if strcmp(ann,'Yes')
                        addi=inputdlg('Enter number of cycles to be
added');
                        if isempty(addi)
                            return
                        end
                        maxcy=maxcy+str2num(cell2mat(addi));
                    end
                end
% Check convergence
                counter=counter+1;
                pcheck=PC./P;pcheck(arrayfun(@isnan,pcheck))=0;
                ucheck=Uc./U;ucheck(arrayfun(@isnan,ucheck))=0;
                vcheck=Vc(2:end-1,:)./V(2:end-
1,:);vcheck(arrayfun(@isnan,vcheck))=0;
                if all(all(abs(pcheck)<accu)) &&
all(all(abs(ucheck)<accu)) ...
                        && all(all(abs(vcheck)<(accu)))

                   break
                end
                if any(isnan(U))
                    nancheck=1;
                    break
                end
                Uold=U;
                Vold=V;
        end
    %%
    if nancheck==1
        msgbox({'Divergence occured, calculation will stop
now.';'';'Reduce the relaxation factors and retry'},'Information');
        vh=findobj('Type','Figure','Tag','vplot');
        close(vh)
        return
    end
    if counter==maxcy
        ex=questdlg('Solution is not converged. Do you want to
proceed?','Warning!','Yes','No','Yes');
        if strcmp(ex,'No')
            return
        end
    end
% background contour plot selection
    if pp==1
        CM=[Pi;avg(P);Po];
        CM=[CM(:,1) avg(CM')' CM(:,end)]/1000;
        cbt='Fluid Pressure (kPa)';
    elseif pp==2
```

```
        CM=(avg([avg(ev(:,1))' eu avg(ev(:,end))']')')*1000;
        cbt='Aperture (mm)';
    end
%output flow rate calculation
         Uend=U(end,:);
         Q=0;
         for i=1:size(Uend,2)
             Q=Q+eu(end,i)*hy*U(end,i);
         end
        %% plotting
          fgf=findobj('Type','figure','Tag','vplot');
          allf=findall(fgf,'Type','Axes');
          if size(allf,1)==2
              ph=allf(2);
          else
              ph=allf;
          end
          %ph=handles.axes2;
          reset(ph);
          [bb,aa]=meshgrid(y,x);
          if direc==2
            [~,hc]=contourf(ph,bb,flipdim(aa,1),CM,20,'w-');
          else
            [~,hc]=contourf(ph,aa,bb,CM,20,'w-');
          end
          colormap(jet);
          set(hc,'EdgeColor','none'); % hc for CONTOURGROUP object
          hold on
          Uee = avg(U);
          Vee = avg([V(2:end-1,1)*0 V(2:end-1,:) V(2:end-
1,end)*0]')';
          zind=find(Uee~=0 & Vee~=0);
          Len = sqrt(Uee.^2+Vee.^2+eps);
          if direc==2
            [y2p,x2p ]=meshgrid(avg(y),avg(x));
            quiver(ph,y2p,flipdim(x2p,1),Vee,-Uee,4,'k-
','LineWidth',0.2,'MaxHeadSize',0.4);
          else
             [y2p, x2p]=meshgrid(avg(y),avg(x));
             quiver(ph,x2p(zind),y2p(zind),Uee(zind),Vee(zind),4,'k-
','LineWidth',0.2,'MaxHeadSize',0.4);
          end
          hold off;
          axis equal;
          if direc==2
              axis([0 ly 0 lx]);
              xt=ly;
              yt=lx;
          else
              axis([0 lx 0 ly]);
              xt=lx;
              yt=ly;
          end
          if cnc==1
            Pcol=[1e-20 max(max(CM))]';
          end
          caxis(Pcol)
          pwr=floor(log10(Q));
          nbr=Q/10^pwr;
```

```
          title(sprintf('Inlet Pressure = %6.1f (kPa), Confining
Pressure = %6.1f (kPa), Flow rate = %6.3f ×10^{%1.0f} (m^3s^{-
1})',inletp/1000,conf/1000,nbr,pwr))
          xlabel('Sample Length (m)');
          ylabel('Sample Width (m)');
          set(gca,'XTick',0:0.01:xt);
          set(gca,'YTick',0:0.01:yt);
          set(gca,'TickDir','Out','Box','Off');

          cb=colorbar;
          title(cb,cbt);
          drawnow
          flo(cnc)=Q;
          cnfs(cnc)=conf;
          ape(cnc)=mean2(e);
          anm=num2str(conf);
    if cnc==1
        cg=get(handles.check1,'Value');
        if cg
            fnm=strcat('C:\Users\ckw172\SkyDrive\Experiment\TAP
sols\MG\',num2str(inletp/1000),'_',num2str(conf/1000),'.jpg');
            haa=findobj('Type','Figure','Tag','vplot');
            print(haa,'-djpeg',fnm,'-r1200');
            incre=inputdlg('Enter confining stress increment
interval in Pa','Information');
            if isempty(incre)
                return
            end
            incre=str2num(cell2mat(incre));
            inum=inputdlg('Enter number of
increments','Information');
            if isempty(inum)
                return
            end
            inum=str2num(cell2mat(inum));
            conf=confini+incre*cnc;
            effsigman=conf-P;
            dnt=(effsigman.*emaxc)./(effsigman+ki*emaxc);
            dntc=dnt-dntp;
            dntp=dnt;
            e=e-dntc;
            e(e<mie)=zer;%e(e<mie)
            SU=zeros(nx+1,ny,inum);
            SV=zeros(nx+2,ny-1,inum);
            SUE=zeros(nx,ny,inum);
            SE=zeros(nx,ny,inum);
            SP=zeros(nx,ny,inum);
            SU(:,:,cnc)=U;
            SV(:,:,cnc)=V;
            SUE(:,:,cnc)=Len;
            SE(:,:,cnc)=e;
            SP(:,:,cnc)=P;
            cnc=cnc+1;
        else
            SU=U;
            SV=V;
            SUE=Len;
            SE=e;
            SP=P;
```

```
            return
        end
    elseif inum==cnc
            SU(:,:,cnc)=U;
            SV(:,:,cnc)=V;
            SUE(:,:,cnc)=Len;
            SE(:,:,cnc)=e;
            SP(:,:,cnc)=P;
            cnc=0;
            ph=findobj('type','Figure','Tag','graph1');
            if isempty(ph)
                hgload('Graph1.fig');
            end
            fh=findobj('type','Axes','Tag','g1');
            if isempty(fh)
              fh=gca;
            end
            reset(fh);
            plot(fh,cnfs,flo);
            title(sprintf('Flow Rate vs Confining Pressure'))
            xlabel('Confining Pressure (Pa)')
            ylabel('Flow Rate (m^3/s)')

    else
       cg=get(handles.check1,'Value');
            conf=confini+incre*cnc;
            effsigman=conf-P;
            dnt=(effsigman.*emaxc)./(effsigman+ki*emaxc);
            dntc=dnt-dntp;
            dntp=dnt;
            e=e-dntc;
            e(e<mie)=zer;%e(e<mie)
            SU(:,:,cnc)=U;
            SV(:,:,cnc)=V;
            SUE(:,:,cnc)=Len;
            SE(:,:,cnc)=e;
            SP(:,:,cnc)=P;
            cnc=cnc+1;
    end
    end
```

SIMPLE algorithm for solving the model assuming contacts are very small apertures is shown below.

A.3.2 *SIMPLE(handles)*

```
function ImplicitSIMPLE(handles)
    global lx ly hx hy nx ny den meu inletp outletp conf ...
           x y  maxcy cnfs flo ape pp epath accu U V confini ...
           SE SP INP g prelax vrelax nthb sthb ki tetax tetay ...
           SU  SV SUE direc
%% Initialization
    getcoeff(handles) % copy user inputs
    w=exist(epath,'file');
```

```
    if w==0
        msgbox('Error reading file' ,'Error');
        return
    end
    inie=load(epath);
    inie(inie==0)=1e-10;
    if direc==2                % transverse direction flow
        [onx,ony]=size(inie);
        inie=rot90(inie,1);
        tm=lx;lx=ly;ly=tm;
        nx=round(nx*ony/onx);
    end
    [e, hx, hy, ny]=descret(inie,lx,ly,nx);
    INP.inaper=e;
    emaxc=e;
    mie=e*0.01;
    zer=1e-10;
    [eu, ev]=center2uv(e);
    x = linspace(0,lx,nx+1);
    y = linspace(0,ly,ny+1);
% Variable initialization
    P=zeros(nx,ny);
    Pxi=(outletp-inletp)/lx;
%initial pressure guess assuming uniform pressure gradient
    for kk=1:nx;
       P(kk,:)=inletp+Pxi*(2*kk-1)*lx/(2*nx);
    end
    U=zeros(nx+1,ny);%-eu.^2*Pxi/12/meu*0.1;
    V = zeros(nx+2,ny-1);
%% Initial boundary conditions
    Pi=zeros(ny,1)'+inletp;
    Po=zeros(ny,1)'+outletp;
    zx=sin(tetax);
    zy=sin(tetay);
% Plotting figure generation
    vvh=hgload('vvectorplot.fig');
    set(0,'Units','Pixels');
    sc=get(0,'ScreenSize');
    mh=findobj('Type','Figure','Tag','main');
    set(vvh,'Units','Pixels');
    set(mh,'Units','Pixels');
    mp=get(mh,'OuterPosition');
    vp=get(vvh,'OuterPosition');
    np=[mp(1)+mp(3)+5,sc(4)-vp(4)-25,vp(3),vp(4)];
    set(vvh,'OuterPosition',np);
% Initial deformation
    dntp=e*0;
    effsigman=conf-P;
    dnt=(effsigman.*emaxc)./(effsigman+ki*emaxc);
    dntc=dnt-dntp;
    dntp=dnt;
    e=e-dntc;
    echeck=e-mie;
    e(echeck<=0)=zer;
% Calculation commences...
%% while loop for confining pressure changing
    cnc=1;
    cnfs=0;flo=0;ape=0;
    while cnc~=0    %cnc counts number of conf. stress increments
```

```
% while loop for solution of one confining pressure
        counter=1;
        nancheck=0;
        Uold=0;
        Vold=0;
        while counter<maxcy
%
%% Solve momentum equation for intermediate velocities
% eq (5.54), (5.55)
            vcou=0;
            [eu, ev]=center2uv(e);
            eub1=[2*eu(1,:)-eu(2,:);eu;2*eu(end,:)-eu(end-1,:)];
            eub=[2*eub1(:,1)-eub1(:,2) eub1 ...
                2*eub1(:,end)-eub1(:,end-1)];
            evb1=[2*ev(1,:)-ev(2,:);ev;2*ev(end,:)-ev(end-1,:)];
            evb=[2*evb1(:,1)-evb1(:,2) evb1 ...
                2*evb1(:,end)-evb1(:,end-1)];
            minape=min(min(e));
            eub(eub<minape)=minape;
            evb(evb<minape)=minape;
            while vcou<maxcy
      % boundary matrices
                if nthb==1
                    Un=[0;U(:,end)*0;0];    %0
                    Vn=V(:,end)*0;    %0
                elseif nthb==2
                    Un=[U(1,end);U(:,end);U(end,end)];
                    Vn=V(:,end);
                end
                if sthb==1
                    Us=[0;U(:,1)*0;0];      %0
                    Vs=V(:,1)*0;      %0
                elseif sthb==2
                    Us=[U(1,1);U(:,1);U(1,end)];
                    Vs=V(:,1);
                end
                UB1=[U(1,:);U;U(end,:)];
                UB=[Un UB1 Us];
                VB1=[V(1,:);V;V(end,:)];
                VB=[VB1(:,1)*0 VB1 VB1(:,end)*0];
                PB1=[Pi;P;Po];
                DPX=[(PB1(2,:)-PB1(1,:))/(hx/2);diff(P)/hx; ...
                    (PB1(end,:)-PB1(end-1,:))/(hx/2)];
                DPY=diff(PB1')'/hy;
      % neighbour U
                UiMj=UB(1:end-2,2:end-1);
                UiPj=UB(3:end,2:end-1);
                UijM=UB(2:end-1,1:end-2);
                UiJP=UB(2:end-1,3:end);
      % neighbour e at U
                euim=eub(1:end-2,2:end-1);
                euip=eub(3:end,2:end-1);
                euJm=eub(2:end-1,1:end-2);
                euJp=eub(2:end-1,3:end);
      % neighbour V
                VIMj=VB(1:end-2,2:end-1);
                VIPj=VB(3:end,2:end-1);
                VIjM=VB(2:end-1,1:end-2);
                VIjP=VB(2:end-1,3:end);
```

```
        % neighbour e at V
                  evIm=evb(1:end-2,2:end-1);
                  evIp=evb(3:end,2:end-1);
                  evjm=evb(2:end-1,1:end-2);
                  evjp=evb(2:end-1,3:end);
        % RHS
                  BiJ=-12*meu./eu-2*meu*eu/hx^2-2*meu*eu/hy^2;
                  BIj=-12*meu./ev-2*meu*ev/hx^2-2*meu*ev/hy^2;
        %
                  UNEW=-(+(den*g*eu*zx)-eu.*DPX)./BiJ ...
                        -((meu*euim/hx^2).*UiMj)./BiJ...
                        -((meu*euip/hx^2).*UiPj)./BiJ...
                        -((meu*euJm/hy^2).*UijM)./BiJ...
                        -((meu*euJp/hy^2).*UiJP)./BiJ;
        %
                  VNEW=-(+(den*g*ev*zy)-(ev.*DPY))./BIj...
                       -((meu*evIm/hx^2).*VIMj)./BIj...
                       -((meu*evIp/hx^2).*VIPj)./BIj...
                       -((meu*evjm/hy^2).*VIjM)./BIj...
                       -((meu*evjp/hy^2).*VIjP)./BIj;
                  vcou=vcou+1;
                  if  max(max(abs((U*(1-vrelax)+UNEW*vrelax-U)./ ...
                    (U+eps))))<accu && max(max(abs((V*(1-vrelax) ...
                     +VNEW*vrelax-V)./(V+eps))))<accu
                      break
                  end
                  if any(isnan(U))
                      nancheck=1;
                      break
                  end
                  if vcou==(maxcy-1)
                      uerr=num2str(max(max(abs((U-U*(1-vrelax) ...
                          +UNEW*vrelax)./U))));
                      verr=num2str(max(max(abs((V-V*(1-vrelax) ...
                          +VNEW*vrelax)./V))));
                      qstr={'Maximum cycles reached in velocity
calculation. Do you want to increase maximum cycles?';'';strcat('
Maximum errors: U = ',uerr,', V = ',verr)};
                      ann=questdlg(qstr,'Edit','Yes','No','Yes');
                      if strcmp(ann,'Yes')
                          addi=inputdlg('Enter number of cycles to be
added');
                          if isempty(addi)
                              return
                          end
                          maxcy=maxcy+str2num(cell2mat(addi));
                      end
                  end
                  U=U*(1-vrelax)+UNEW*vrelax;
                  V=V*(1-vrelax)+VNEW*vrelax;
              end
              if nancheck==1
                  msgbox({'Divergence occured, calculation will stop
now.';'';'Select number of nodes so that a cell is close to a
square'},'Information');
                  vh=findobj('Type','Figure','Tag','vplot');
                  close(vh)
                  return
              end
```

```
%% Solve pressure correction. Equation (5.68)
            pcou=0;
            PC=P*0;
% coefficient matrices
                BiJp=-12*meu./eu-2-meu*eu/hx^2-meu*eu/hy^2;
                BIjp=-12*meu./ev-2-meu*ev/hx^2-meu*ev/hy^2;
                BIjBp=[BIjp(:,1) BIjp BIjp(:,end)];
                MiJ=-(den*eu(2:end,:).^2)./(BiJp(2:end,:)*hx^2);
                MimJ=-(den*eu(1:end-1,:).^2)./(BiJp(1:end-
1,:)*hx^2);
                MIj=-(den*evb(3:end-2,2:end).^2)./(BIjBp(2:end-
1,2:end)*hy^2);
                MIjm=-(den*evb(3:end-2,1:end-1).^2)./(BIjBp(2:end-
1,1:end-1)*hy^2);
                MIJ=-MiJ-MimJ-MIj-MIjm;
% RHS
                UX=den*e.*(diff(U)/hx);
                VY=den*e.*(diff(([Vn(2:end-1) V(2:end-1,:) Vs(2:end-
1)])')'/hy);

            while pcou<maxcy
        % boundary matrix
                PCB1=[Pi*0;PC;Po*0];
                PCB=[PCB1(:,1) PCB1 PCB1(:,end)]; %dp/dn=0 at wall
        %
                PCNEW=-(MiJ.*PCB(3:end,2:end-1))./MIJ-
(MimJ.*PCB(1:end-2,2:end-1))./MIJ...
                      -(MIj.*PCB(2:end-1,3:end))./MIJ-
(MIjm.*PCB(2:end-1,1:end-2))./MIJ...
                      +UX./MIJ+VY./MIJ;
                pcou=pcou+1;
                if all(abs((PC*(1-prelax)+PCNEW*prelax-
PC)./(PC+eps))<accu)
                    break
                end
                PC=PC*(1-prelax)+PCNEW*prelax;
            end
        %Pressure correction
                PC=PC*prelax;
                P=P+PC;
        %Velocity correction
                Pcu=[((PC(1,:)-Pi*0)/hx);diff(PC(1:end,:))/hx;(Po*0-
PC(end,:))/hx];

Pcv=[diff(PC(1,:)/2)/hy;diff(PC')'/hy;diff(PC(end,:)/2)/hy];
                Uc=eu.*Pcu./BiJ;
                Vc=ev.*Pcv./BIj;
                U=prelax*Uc+U;
                V=prelax*Vc+V;
                %
                set(handles.text33,'String',num2str(counter));
                %
                if counter==(maxcy-1)
                     uerr=num2str(max(max(abs((U-Uold)./U))));
                    verr=num2str(max(max(abs((V-Vold)./V))));
                    perr=num2str(max(max(abs(PC./P))));
                    qstr={'Maximum cycles reached. Do you want to
increase maximum cycles?';'';strcat(' Maximum errors: U = ',uerr,',
V = ',verr,', P = ',perr)};
```

```
                    ann=questdlg(qstr,'Edit','Yes','No','Yes');
                    if strcmp(ann,'Yes')
                        addi=inputdlg('Enter number of cycles to be
added');
                        if isempty(addi)
                            return
                        end
                        maxcy=maxcy+str2num(cell2mat(addi));
                    end
                end
% Check convergence
                counter=counter+1;
                if all(all(abs(PC./P)<accu)) &&
all(all(abs(Uc./U)<accu)) ...
                        && all(all(abs(Vc(2:end-1,:)./V(2:end-
1,:))<(accu)))

                   break
                end
                if any(isnan(U))
                    nancheck=1;
                    break
                end
                Uold=U;
                Vold=V;
        end
    %%
    if nancheck==1
        msgbox({'Divergence occured, calculation will stop
now.';'';'Reduce the relaxation factors and retry'},'Information');
        vh=findobj('Type','Figure','Tag','vplot');
        close(vh)
        return
    end
    if counter==maxcy
        ex=questdlg('Solution is not converged. Do you want to
proceed?','Warning!','Yes','No','Yes');
        if strcmp(ex,'No')
            return
        end
    end
% background contour plot selection
    if pp==1
        CM=[Pi;avg(P);Po];
        CM=[CM(:,1) avg(CM')' CM(:,end)]/1000;
        cbt='Fluid Pressure (kPa)';
    elseif pp==2
        CM=(avg([avg(ev(:,1))' eu avg(ev(:,end))']')')*1000;
        cbt='Aperture (mm)';
    end
%output flow rate calculation
         Uend=U(end,:);
         Q=0;
         NCoutA=0;
         for i=1:size(Uend,2)
             Q=Q+eu(end,i)*hy*U(end,i);
             NCoutA=NCoutA+eu(end,i)*hy;
         end
         NCoutA
```

```
          eAVE=mean2(e)
          eRMS=sqrt(mean2(e.^2))
         %% plotting
           fgf=findobj('Type','figure','Tag','vplot');
           allf=findall(fgf,'Type','Axes');
           if size(allf,1)==2
               ph=allf(2);
           else
               ph=allf;
           end
           %ph=handles.axes2;
           reset(ph);
           [bb,aa]=meshgrid(y,x);
           if direc==2
             [~,hc]=contourf(ph,bb,flipdim(aa,1),CM,20,'w-');
           else
             [~,hc]=contourf(ph,aa,bb,CM,20,'w-');
           end
           hold on
           colormap(jet);
           set(hc,'EdgeColor','w'); % hc for CONTOURGROUP object
'none' removes the edge color
           Uee = avg(U);
           Vee = avg([V(2:end-1,1)*0 V(2:end-1,:) V(2:end-
1,end)*0]')';
           zind=find(Uee~=0 & Vee~=0);
           Len = sqrt(Uee.^2+Vee.^2+eps);
           if direc==2
             [y2p,x2p ]=meshgrid(avg(y),avg(x));
             quiver(ph,y2p,flipdim(x2p,1),Vee,-Uee,6,'k-
','LineWidth',0.2,'MaxHeadSize',0.4);
           else
              [y2p, x2p]=meshgrid(avg(y),avg(x));
              quiver(ph,x2p(zind),y2p(zind),Uee(zind),Vee(zind),6,'k-
','LineWidth',0.2,'MaxHeadSize',0.4);
           end
           hold off;axis equal;
           if direc==2
               axis([0 ly 0 lx]);
               xt=ly;
               yt=lx;
           else
               axis([0 lx 0 ly]);
               xt=lx;
               yt=ly;
           end

            if cnc==1
                Pcol=(0:0.0005:max(max(CM)))';
            end
           caxis(Pcol([1 end]))
           pwr=floor(log10(Q));
           nbr=Q/10^pwr;
           title(sprintf('Inlet Pressure = %6.1f (kPa), Confining
Pressure = %6.1f (kPa), Flow rate = %6.3f ×10^{%1.0f} (m^3s^{-
1})',inletp/1000,conf/1000,nbr,pwr))
           xlabel('Sample Length (m)');
           ylabel('Sample Width (m)');
           set(gca,'XTick',0:0.01:xt);
```

```
        set(gca,'YTick',0:0.01:yt);
        set(gca,'TickDir','Out','Box','Off');
        colormap(jet);
```

Beyond this point the codes are same as for the SIMPLEwithcontacts(handles). For the SIMPLEC algorithm, the pressure correction codes should be changed and that section only is shown below.

SIMPLEC pressure correction

```
%%
        % coefficient matrices
        BiJp=-12*meu./eu-2-meu*eu/hx^2-meu*eu/hy^2;
        BIjp=-12*meu./ev-2-meu*ev/hx^2-meu*ev/hy^2;
        BIjBp=[BIjp(:,1) BIjp BIjp(:,end)];
        MiJ=-(den*eu(2:end,:).^2)./(BiJp(2:end,:)*hx^2);
        MimJ=-(den*eu(1:end-1,:).^2)./(BiJp(1:end-1,:)*hx^2);
        MIj=-(den*evb(3:end-2,2:end).^2)./(BIjBp(2:end-
1,2:end)*hy^2);
        MIjm=-(den*evb(3:end-2,1:end-1).^2)./(BIjBp(2:end-1,1:end-
1)*hy^2);
        MIJ=-MiJ-MimJ-MIj-MIjm;
        % RHS
        UX=den*e.*(diff(U)/hx);
        VY=den*e.*(diff(([Vn(2:end-1) V(2:end-1,:) Vs(2:end-
1)])')'/hy);

    while pcou<maxcy
        % boundary matrix
        PCB1=[Pi*0;PC;Po*0];
        PCB=[PCB1(:,1) PCB1 PCB1(:,end)]; %dp/dn=0 at wall
        %
        PCNEW=-(MiJ.*PCB(3:end,2:end-1))./MIJ-(MimJ.*PCB(1:end-
2,2:end-1))./MIJ...
              -(MIj.*PCB(2:end-1,3:end))./MIJ-(MIjm.*PCB(2:end-
1,1:end-2))./MIJ...
              +UX./MIJ+VY./MIJ;
        pcou=pcou+1;
        if all(abs((PC*(1-prelax)+PCNEW*prelax-PC)./(PC+eps))<accu)
            break
        end
        PC=PC*(1-prelax)+PCNEW*prelax;
        %PC=PCNEW;
    end
```

Other supporting functions described in Chapter 4 are shown below.

A.3.3 *getcoeff(handles)*

```
function getcoeff(handles)
global lx ly den meu inletp outletp conf tetax tetay g prelax vrelax
global maxcy nthb sthb mtime dt epath accu INP confini nx ny ki
lx=str2num(get(handles.l,'String'));
ly=str2num(get(handles.w,'String'));
den=str2num(get(handles.den,'String'));
meu=str2num(get(handles.meu,'String'));
inletp=str2num(get(handles.ip,'String'));
outletp=str2num(get(handles.op,'String'));
conf=str2num(get(handles.cnf,'String'));
confini=str2num(get(handles.cnf,'String'));
tetax=degtorad(str2num(get(handles.tx,'String')));
tetay=degtorad(str2num(get(handles.ty,'String')));
g=str2num(get(handles.g,'String'));
prelax=str2num(get(handles.prl,'String'));
vrelax=str2num(get(handles.vrl,'String'));
maxcy=str2num(get(handles.mcy,'String'));
nthb=get(handles.uby,'Value');
sthb=get(handles.vby,'Value');
mtime=str2num(get(handles.etime,'String'));
dt=str2num(get(handles.estep,'String'));
accu=str2num(get(handles.accu,'String'));
epath=get(handles.edit33,'String');
nx=str2num(get(handles.nx,'String'));
ny=str2num(get(handles.ny,'String'));
ki=str2num(get(handles.kni,'String'));
INP.lx=lx;
INP.ly=ly;
INP.den=den;
INP.meu=meu;
INP.inletp=inletp;
INP.outletp=outletp;
INP.conf=conf;
INP.tetax=tetax;
INP.tetay=tetay;
INP.g=g;
INP.prelax=prelax;
INP.vrelax=vrelax;
INP.maxcy=maxcy;
INP.nthb=nthb;
INP.sthb=sthb;
INP.mtime=mtime;
INP.dt=dt;
INP.accu=accu;
INP.epath=epath;
INP.ki=ki;


function rad=degtorad(deg)
rad=deg*pi()/180;
```

A.3.4 *descret(inie,lx,ly,nx)*

```
function [newe, hx, hy, ny]=descret(e,lx,ly,nx)
% DESCRET returns the aperture matrix interpolated according to the
%    user specified number of cells in length direction assuming
%    cubic distribution. The descretized domaindetails (cell height
%    and width and cells in width direction) are also returns.


[onx,ony]=size(e);
xgv = linspace(0,lx,onx);
ygv = linspace(0,ly,ony);
[X,Y] = ndgrid(xgv,ygv);
F = griddedInterpolant(X, Y, e, 'cubic');
xqgv=linspace(0,lx,nx);
ny=round(nx*ony/onx);
yqgv = linspace(0,ly,ny);
[Xq,Yq] = ndgrid(xqgv,yqgv);
newe=F(Xq, Yq);
hx=lx/nx;
hy=ly/ny;
```

A.3.5 *avg(A,k)*

```
function B = avg(A,k)
if nargin<2, k = 1; end
if size(A,1)==1, A = A'; end
if k<2, B = (A(2:end,:)+A(1:end-1,:))/2; else B = avg(A,k-1); end
if size(A,2)==1, B = B'; end
end
```

A.3.6 *center2uv(A)*

```
function [Au,Av]=center2uv(A)
%e and rho at faces for velocity cells
mine=min(min(A));
 Au=[3/2*A(1,:)-A(2,:)/2;avg(A);3/2*A(end,:)-A(end-1,:)/2];
 Av=[3/2*avg(A(1,:)')-avg(A(2,:)')/2;avg(A')';3/2*avg(A(end,:)')-
avg(A(end-1,:)')/2];
 Au(Au<mine)=mine;
 Av(Av<mine)=mine;
```

A.3.7 *localb(e)*

```
function [uLid, uRid, vDid, vUid]=localb(e)
% LOCALB returns the indices of velocity components became zero due
%     to the zero apertures or local boundaries. At the faces
```

```
%     of a contact (zero aperture) the velocities are made
%     zero when no slip boundaries are assumed.
[nx,ny]=size(e);
[erowi,ecoli]=find(e==0);
uLrowi=erowi;uLcoli=ecoli;
%uLrowi can't have 0 or less
uLcoli(uLrowi<=0)=[];
uLrowi(uLrowi<=0)=[];
uLid=sub2ind([nx+1 ny],uLrowi,uLcoli);
%
uRrowi=erowi+1;uRcoli=ecoli;
%uRrowi can't have nx+2 or greater
uRcoli(uRrowi>=nx+2)=[];
uRrowi(uRrowi>=nx+2)=[];
uRid=sub2ind([nx+1 ny],uRrowi,uRcoli);
%
vDrowi=erowi+1;vDcoli=ecoli-1;
%vDcoli can't have 0 or less
vDrowi(vDcoli<=0)=[];
vDcoli(vDcoli<=0)=[];
vDid=sub2ind([nx+2 ny-1],vDrowi,vDcoli);
%
vUrowi=erowi+1;vUcoli=ecoli;
%vUcoli can't have ny or greater
vUrowi(vUcoli>=ny)=[];
vUcoli(vUcoli>=ny)=[];
vUid=sub2ind([nx+2 ny-1],vUrowi,vUcoli);
```

A.3.8 *pneibr(PCB)*

```
function [PIpJ,PImJ,PIJp,PIJm]=pneibr(PCB)
% PNEIBR returns the four neighbour matrices of a particular
variable
%     in the domain assuming 0 entries are non-existing and local
%     boundaries. The derivatives of the variable perpendicular to
the local
%     boundaries are taken as zero.

%dp/dn=0 at walls of ocal boundaries
[k,l]=find(PCB==0); %k l contains indices for zeros
% for I+1
    ipx=k;kpx=k;
    ipy=l;lpy=l;
    ipy(k==size(PCB,1))=[]; %omit last row zeros
    ipx(k==size(PCB,1))=[];
    ipx=ipx-1;
    ipx(ipx==0)=1; %sustitute zero index 1st row
    ipD=sub2ind(size(PCB),ipx,ipy);
    PCBx=PCB;
    lpy(kpx==size(PCB,1))=[]; %omit last row zeros
    kpx(kpx==size(PCB,1))=[];
    in=sub2ind(size(PCB),kpx,lpy);
    PCBx(in)=PCBx(ipD);
PIpJ=PCBx(3:end,2:end-1);
% for I-1
    ipx=k;kpx=k;
```

```
    ipy=l;lpy=l;
    ipy(k==1)=[];%omit first row zeros
    ipx(k==1)=[];
    ipx=ipx+1;
    ipx(ipx==(size(PCB,1)+1))=size(PCB,1); %subs I+1 index last row
    ipD=sub2ind(size(PCB),ipx,ipy);
    lpy(kpx==1)=[];
    kpx(kpx==1)=[];
    in=sub2ind(size(PCB),kpx,lpy);
    PCBx=PCB;
    PCBx(in)=PCBx(ipD);
PImJ=PCBx(1:end-2,2:end-1);
% for J+1
    ipx=k;kpx=k;
    ipy=l;lpy=l;
    ipx(l==size(PCB,2))=[];
    ipy(l==size(PCB,2))=[];  % omit last cols
    ipy=ipy-1;
    ipy(ipy==0)=1;      %subs zero index 1st col
    ipD=sub2ind(size(PCB),ipx,ipy);
    kpx(lpy==size(PCB,2))=[];
    lpy(lpy==size(PCB,2))=[]; %omit last col zeros
    in=sub2ind(size(PCB),kpx,lpy);
    PCBx=PCB;
    PCBx(in)=PCBx(ipD);
PIJp=PCBx(2:end-1,3:end);
% for J-i
    ipx=k;kpx=k;
    ipy=l;lpy=l;
    ipy=ipy+1;
    ipx(l==1)=[];
    ipy(l==1)=[]; %omit first col zeros
    ipy(ipy==(size(PCB,2)+1))=size(PCB,2);      %subs zero index
last col
    ipD=sub2ind(size(PCB),ipx,ipy);
    kpx(lpy==1)=[];
    lpy(lpy==1)=[]; %omit first col zeros
    in=sub2ind(size(PCB),kpx,lpy);
    PCBx=PCB;
    PCBx(in)=PCBx(ipD);
PIJm=PCBx(2:end-1,1:end-2);
```

A.3.9 Createaperture(elwr,eupr,lx,trim)

```
function e=createaperture(elwr,eupr,lx,trim)
%% CREATEAPERTURE returns the aperture matrix for given upper and
lower
%    surface profiles of a fracture. The surface profiles might have
%    different origins but should have same number of rows and
columns of
%    data. The rows should represent the length and positive trim
refers
%    shifting the upper profile forward and vise versa.

if size(elwr)~=size(eupr)
    e='Surface profiles are not compatible';
```

```
    return
end
[nx,~]=size(elwr);
if trim~=0
    t=nx*round(trim/lx);
    if trim>0
        eupr=eupr(1:(end-t),:);
        elwr=elwr((t+1):end,:);
    elseif trim <0
        eupr=eupr((t+1):end,:);
        elwr=elwr(1:(end-t),:);
    end
end
[nx,ny]=size(elwr);
xhalf=round(nx/2);
yhalf=round(ny/2);
Q1s=elwr(1:xhalf,1:yhalf)+eupr(1:xhalf,1:yhalf);
H1=max(max(Q1s));
Q2s=elwr(1:xhalf,(yhalf+1):end)+eupr(1:xhalf,(yhalf+1):end);
H2=max(max(Q2s));
Q3s=elwr((xhalf+1):end,1:yhalf)+eupr((xhalf+1):end,1:yhalf);
H3=max(max(Q3s));
Q4s=elwr((xhalf+1):end,(yhalf+1):end)+eupr((xhalf+1):end,(yhalf+1):e
nd);
H4=max(max(Q4s));
H=(H1+H2+H3+H4)/4;
fu=H-eupr;
e=fu-elwr;
e(e<0)=0;
```

Appendix B: Graphical User Interfaces of RFFS

B.1 MAIN graphical user inter face

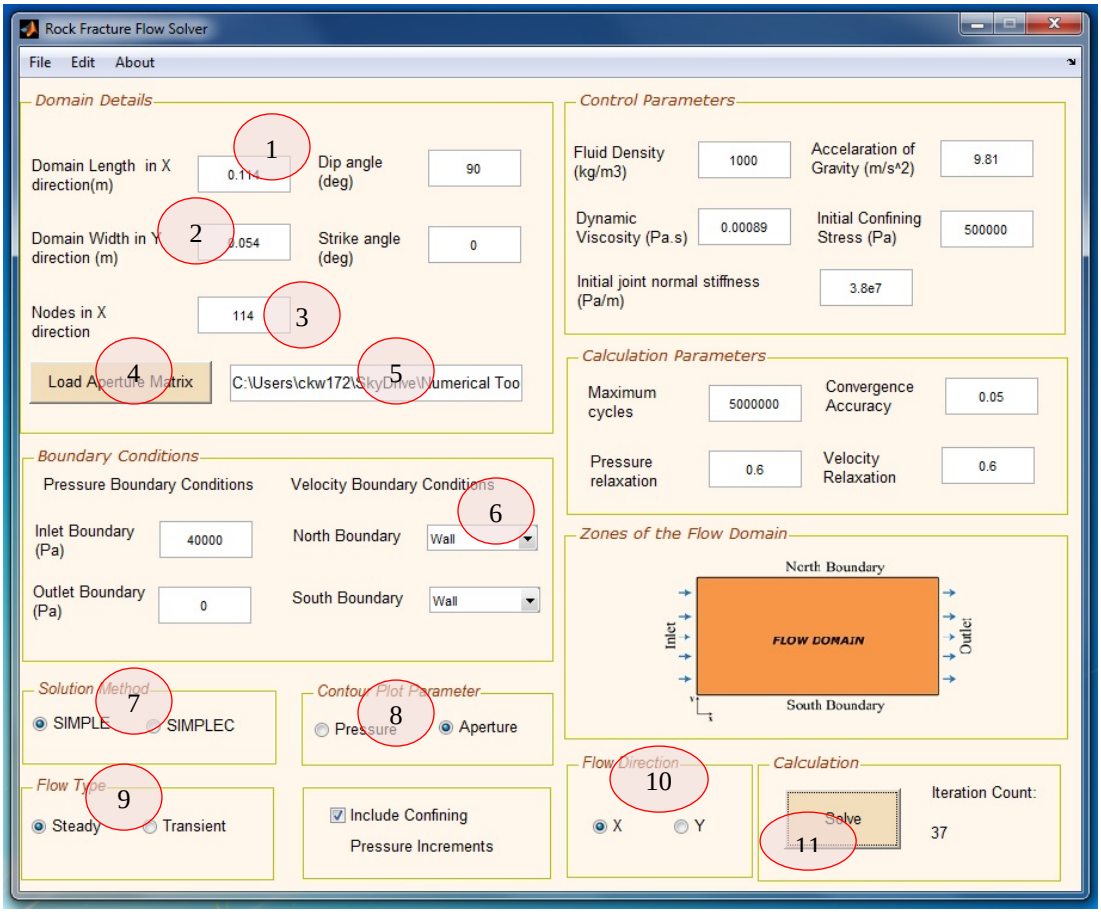


Figure B.1: The main GUI of the programme - #1

Figures B.1 and B.2 show the main graphical user interface and some important items in the GUI are numbered and explained below.

1. Enter the length of the fracture in meters
2. Enter the width of fracture in meters
3. Enter the required number of (nodes) finite volumes in X-direction. Number of nodes in Y-direction is selected according to the aspect ratio of the fracture.
4. Click here to select the data file containing the aperture matrix in a subdirectory
5. Aperture matrix data file address is displayed here once selected. The address can manually be entered
6. Select the type of boundary for the sides of the fracture. If a membrane is assumed as the boundary, 'Wall' can be selected. When 'Mirror' is selected flow can be simulated without the effect of the membrane
7. Select the algorithm, SIMPLE or SIMPLEC
8. Select the contour plot parameter for the velocity vector plot background
9. Select flow type: steady or transient
10. Select flow direction: x – longitudinal, y – transverse
11. Click to start calculation
12. Enter the inclination of fracture. For sub-axial fractures dip angle is 90^0
13. Enter fluid and fracture properties in the given units

14. Enter the accuracy required. 0.01 equals to 1% error margin between two successive solutions. Once all the variables satisfy the condition, solution will stop
15. Enter the maximum cycles to be iterated and calculation is paused at that iteration and user is asked to add additional iterations if needed. Otherwise calculation can be aborted.
16. Enter the relaxation factor. The value should be less than 1 to under relax the solution and lower the relaxation factor, stable the solution but calculation time increases
17. Entering outlet boundary pressure greater than the inlet boundary, reverse flow can be simulated
18. Illustration of transverse flow to identify the boundary names
19. Check this box if confining increments are needed
20. The main iteration loop count is displayed here

Additionally Figure B.3 shows the velocity plot GUI with its data storing options and two instances of error messages are shown in Figures B.4 and B5.

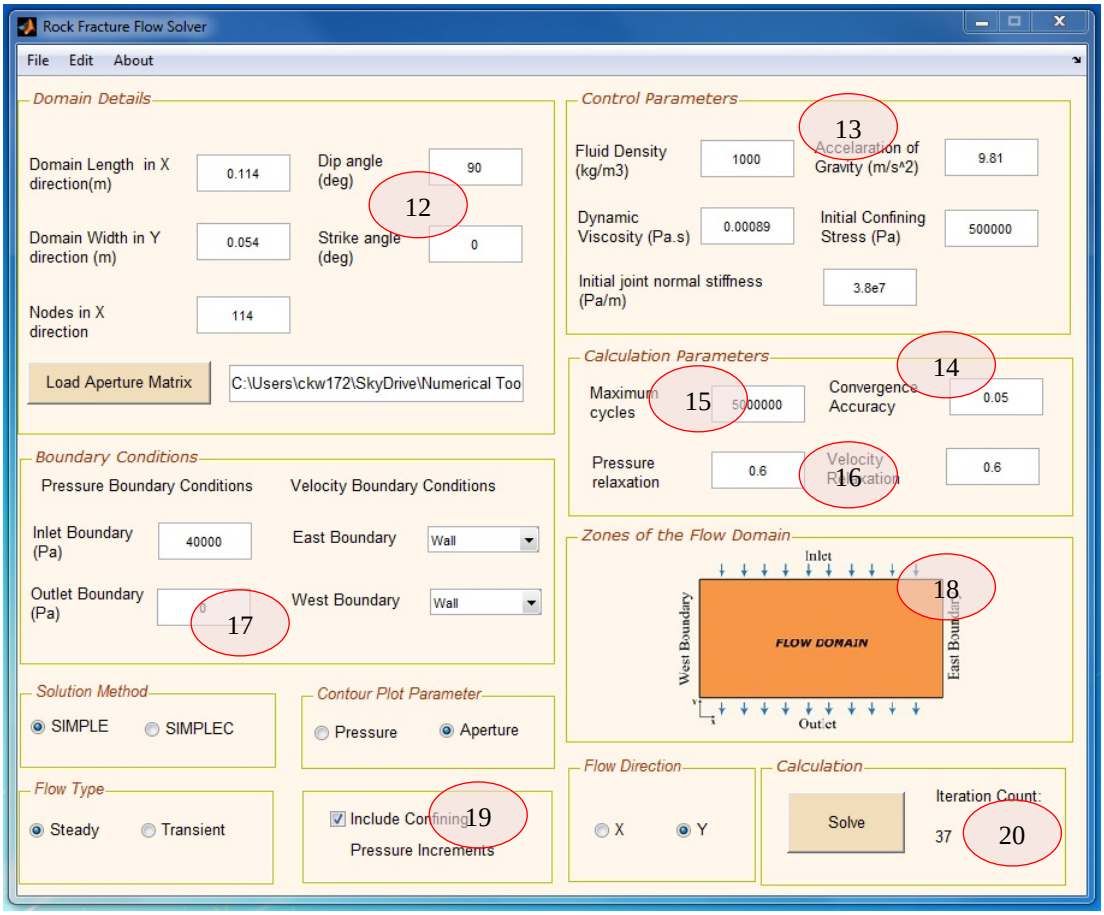


Figure B.2: Main GUI of the programme - #2

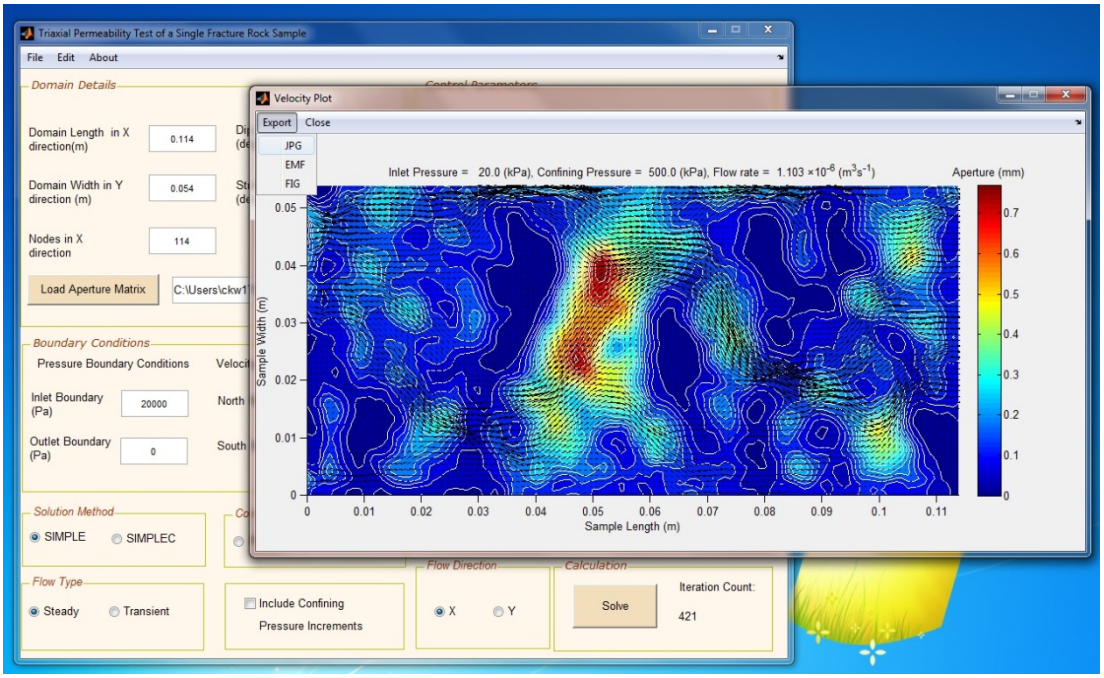


Figure B.3 Velocity vector plot

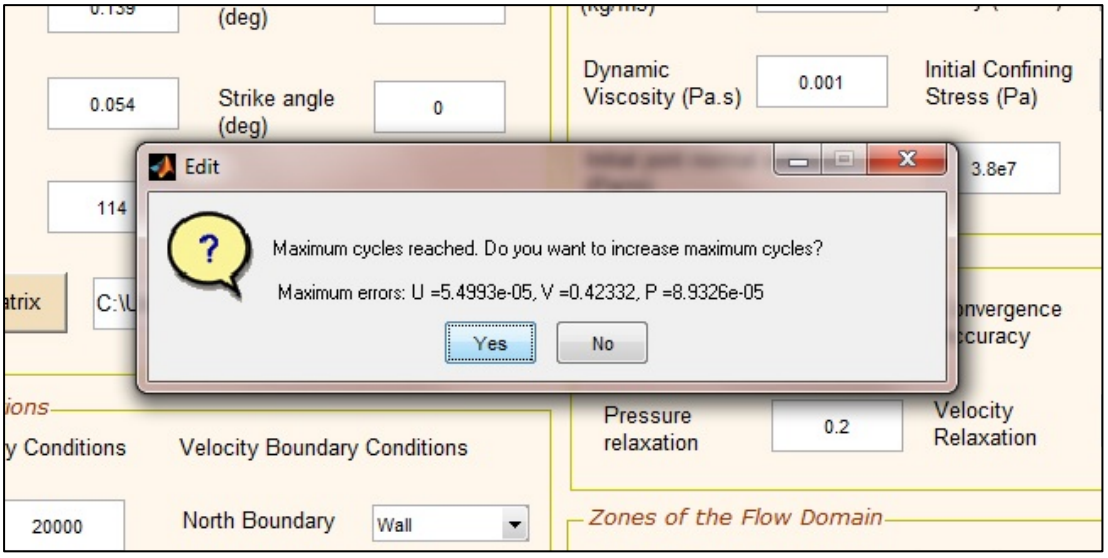


Figure B.4: Notification when maximum cycles are reached

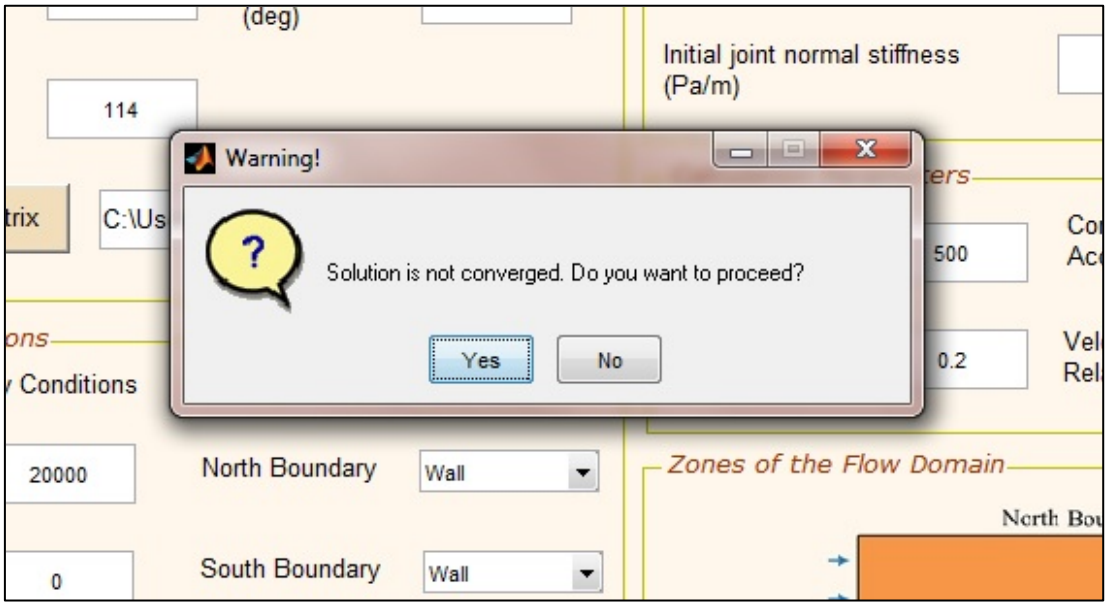


Figure B.4: Notification if solution not converged at the end of preferred iterations

Appendix C: Two-Dimensional Multi Phase Flow Model for Rough Rock Fractures

C.1 Multi-Phase Flow Modelling

In the previous chapter, the flow equations of a single phase were derived applicable to a deformable rock fracture. This chapter extends the flow equations when there is more than one phase flowing in a deformable rock fracture. When additional phases are introduced to the flow domain, new conservation equations are needed to be introduced and the current conservation equations are needed to be modified. Each phase in the flow domain possesses momentum conservation equations and a continuity equation. The different between single phase governing equations and multi-phase governing equations is the continuity equation and momentum conservation equations are multiplied with the volume fraction of the respective phase and the interaction forces among the phases are added to the momentum equations. The summation of all the volume fractions of phases is equal to unity. Additionally, if phase changes and/or chemical reactions are taken place, they also should be added to the momentum conservation as a force and in continuity equation as a source term. In this model those terms are neglected for simplicity. The conservation equations for a phase x can be written as follows.

C.1.1 Continuity equation of a phase of multi-phase flow

$$\frac{\partial(C_x\rho_x)}{\partial t} + \boldsymbol{\nabla}\cdot(C_x\rho_x\boldsymbol{V_x}) = S_x \tag{C.1}$$

C.1.2 Momentum equation of a phase of multi-phase flow domain

$$\frac{\partial(C_x\rho_x\boldsymbol{V_x})}{\partial t} + \boldsymbol{\nabla}.(C_x\rho_x\boldsymbol{V_x}\boldsymbol{V_x}) + C_x\boldsymbol{\nabla}p - C_x\rho_x g\boldsymbol{\nabla}Z - C_x\mu\nabla^2\boldsymbol{V_x} + \sum_{\substack{i=1\\ i\neq x}}^{n} R_{ix} = 0 \tag{C.2}$$

Where,

C_x - Volume fraction of phase x

S_x - Source term of phase x

$\rho_x = C_x\rho_{xp}$ - Effective density of phase x

ρ_{xp} - Physical density of phase x

R_{ix} – Interaction forces between two phases i and x

n - Number of phases

Now the conservation equations are averaged over the direction of aperture height as described in the section 3.2 for a rock joint with single phase flow. The integrated continuity equation is given in the Equation (C.3) and the integrated momentum equation is given in the Equation (C.4).

$$e\frac{\partial(C_x\rho_x)}{\partial t} + \acute{\boldsymbol{\nabla}}\cdot(C_x\rho_x e\langle\boldsymbol{V}\rangle_{\boldsymbol{x}}) = eS_x \tag{C.3}$$

$$\frac{\partial(C_x\rho_x e\overline{\boldsymbol{V}}_{\boldsymbol{x}})}{\partial t} + \acute{\boldsymbol{\nabla}}.(C_x\rho_x e\overline{\boldsymbol{V}}_{\boldsymbol{x}}\overline{\boldsymbol{V}}_{\boldsymbol{x}}) + C_x e\acute{\boldsymbol{\nabla}}p - C_x\rho_x eg\acute{\boldsymbol{\nabla}}Z - C_x\mu\acute{\nabla}^2(e\overline{\boldsymbol{V}}_{\boldsymbol{x}}) + \frac{12C_x\mu}{e}\overline{\boldsymbol{V}}_{\boldsymbol{x}} + \sum_{\substack{i=1 \\ i\neq x}}^{n} eR_{ix} = 0 \quad \text{(C.4)}$$

C.2 Two-phase flow model for fluid-sediment system

Rock fractures contain infilled sediments due the fracture surface erosion and fracture asperity degradation. Two-phase flow of water and sediments is common for rock fractures. The multi-phase flow model can be modified to fluid-sediment flow accordingly. There is no mass transfer in between phases in the case of sediment and water flow. There can be interaction forces among phases and a lift force applied to sediment particles which should appear in the momentum conservation equation. At this instance, those terms are neglected. When eroded sediments are transported with the fluid, a control volume is shared by the two phases.

For sediment and fluid flow system, the source term for the fluid phase will be zero and for the sediment phase, erosion of sediments become the source term. When the volume concentration of the sediments is C_s, the volume fraction of fluid will be $(1 - C_s)$. Now four equations can be obtained to be solved for the flow problem. Both the fluid and solids share the same pressure in this model.

C.2.1 Conservation equations for fluid phase

$$e_h\frac{\partial\big((1-C_s)\rho_f\big)}{\partial t} + \acute{\boldsymbol{\nabla}}\cdot\big((1-C_s)\rho_f e_h\overline{\boldsymbol{V}}_f\big) = 0 \quad \text{(C.5)}$$

$$\frac{\partial\left((1-C_s)\rho_f e_h \overline{\boldsymbol{V}}_f\right)}{\partial t} + \acute{\boldsymbol{\nabla}}.\left((1-C_s)\rho_f e_h \overline{\boldsymbol{V}}_f \overline{\boldsymbol{V}}_f\right) + (1-C_s)e_h \acute{\boldsymbol{\nabla}} p - (1-C_s)\rho_f e_h g \acute{\boldsymbol{\nabla}} Z - (1-C_s)\mu \acute{\nabla}^2\left(e_h \langle \boldsymbol{V} \rangle_f\right) + \frac{12(1-C_s)\mu}{e_h}\langle \boldsymbol{V} \rangle_f = 0 \quad \text{(C.6)}$$

C.2.2 Conservation equations for sediment phase

$$e_h \frac{\partial (C_s \rho_s)}{\partial t} + \acute{\boldsymbol{\nabla}} \cdot (C_s \rho_s e_h \langle \boldsymbol{V} \rangle_{\boldsymbol{s}}) = r_e \quad \text{(C.7)}$$

$$\frac{\partial (C_s \rho_s e_h \langle \boldsymbol{V} \rangle_s)}{\partial t} + \acute{\boldsymbol{\nabla}}.(C_s \rho_s e_h \langle \boldsymbol{V} \rangle_s \langle \boldsymbol{V} \rangle_s) + C_s e_h \acute{\boldsymbol{\nabla}} p - C_s \rho_s e_h g \acute{\boldsymbol{\nabla}} Z - C_s \mu \acute{\nabla}^2 (e_h \langle \boldsymbol{V} \rangle_s) + \frac{12 C_s \mu}{e_h}\langle \boldsymbol{V} \rangle_s = 0 \quad \text{(C.8)}$$

ρ_f and ρ_s are effective densities of fluid and sediments respectively. The equations (C.5), (C.6), (C.7) and (C.8) are solved numerically using finite volume method and SIMPLE algorithm by the software developed in MATLAB programming language.

C.3 Sediment Erosion in Rock Fractures

When a rock fracture contains infill material and fluid flows through it, the infill material may be get eroded. Then the eroded material is transported with the fluid. The erosion of the infill material changes the space available inside the fracture for the fluid flow. This will be influenced by the stress environment too. When modelling this type of flow, the fluid is assumed to be transported on top of the sediment layer as shown in Figure C.1. Erosion of sediments is related to the effective shear stress applied to the sediment surface and the erodibility of sediments. Effective shear stress is the difference between applied shear stress, (τ) and the

critical shear stress, (τ_c) of sediments. If the critical shear stress is equal or higher than the applied shear stress, then the effective shear stress is zero. Erodibility of sediments is given by the coefficient of erosion (α). Then the rate of erosion, (r_e) as follows,

$$r_e = \alpha(\tau - \tau_c) \tag{C.9}$$

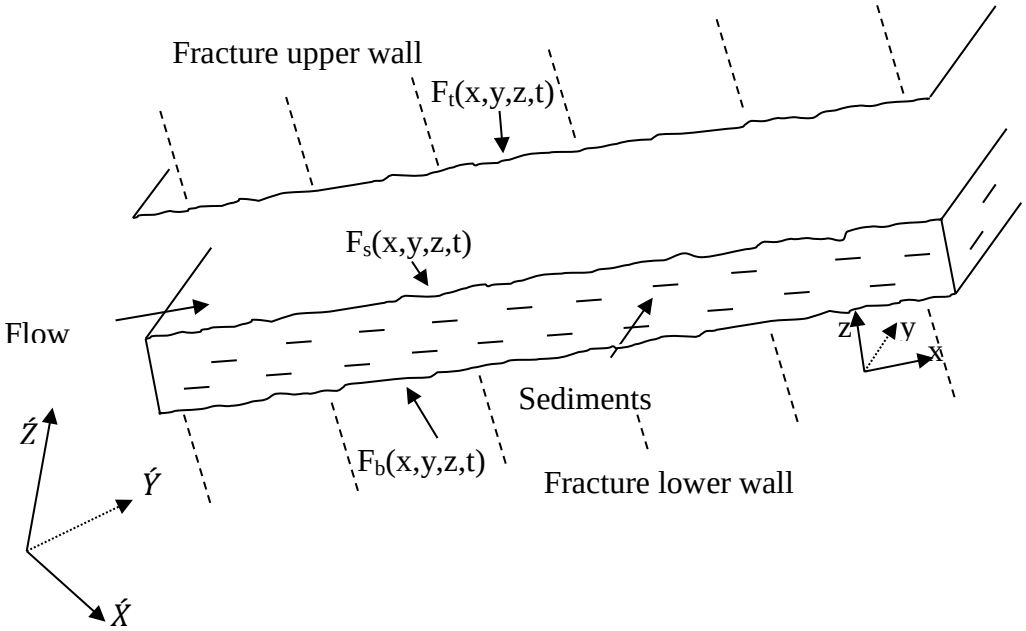


Figure C.1: Rock fracture with infilled sediments

The eroded sediments are transported with the fluid. Transport equation and sediment continuity equation should be solved after the solution of velocity and pressure for the fluid. The eroded sediments in a particular time interval will be the source term of the sediment continuity equation in that time interval. Then the sediment source can be calculated as,

$$S_e = \frac{r_e}{e_h} \tag{C.10}$$

Equation (C.9) measures the rate of erosion. Applied shear stress is a function of velocity and density of the fluid (Price and Indraratna 2005), (Fourar et al. 1993) and is given by ,

$$\tau = \frac{1}{2} C_f \rho V^2 \tag{C.11}$$

C_f is the friction factor. Critical shear stress, (τ_c) in equation (C.9) is a property of the infill material.

C.4 Transient Flow

The applied effective normal stress to the fracture deforms the fracture mechanically. This effect can be coupled with time according to the knowledge about stress environment. Furthermore, when sediments present in rock fracture, the hydraulic aperture depends on the amount of sediments present in the fracture. Apart from that, sediments may get eroded with the flow and hydraulic aperture changes with time. When a flow model is required to address both these phenomenon, transient flow should be considered.

C.4.1 *Discretisation of Governing Equations*

When comes to the transient flow, a finite volume should be considered in a finite time element as the particular control volume's flow data changes with time. To achieve this, the governing equations should be integrated over a control volume and over a small time interval.

C.4.1.1 Momentum equations

$$\int_{t}^{t+\Delta t}\int_{cv}\left[\frac{\partial}{\partial t}(\rho e_h\langle u\rangle)\right]dvdt$$

$$= \int_{t}^{t+\Delta t}\int_{cv}\left[-\frac{\partial}{\partial x}(e_h\rho\langle u\rangle\langle u\rangle)-\frac{\partial}{\partial y}(e_h\rho\langle u\rangle\langle v\rangle)\right.$$

$$+\mu\frac{\partial^2}{\partial x^2}(e_h\langle u\rangle)+\mu\frac{\partial^2}{\partial y^2}(e_h\langle u\rangle)-\frac{12\mu}{e_h}\langle u\rangle+\rho g e_h\frac{\partial \acute{Z}}{\partial x}$$

$$\left.-e_h\frac{\partial p}{\partial x}\right]dvdt \qquad \text{(C.12)}$$

$$\int_{t}^{t+\Delta t}\int_{cv}\left[\frac{\partial}{\partial t}(\rho e_h\langle v\rangle)\right]dvdt$$

$$= \int_{t}^{t+\Delta t}\int_{cv}\left[-\frac{\partial}{\partial x}(e_h\rho\langle v\rangle\langle u\rangle)-\frac{\partial}{\partial y}(e_h\rho\langle v\rangle\langle v\rangle)\right.$$

$$+\mu\frac{\partial^2}{\partial x^2}(e_h\langle v\rangle)+\mu\frac{\partial^2}{\partial y^2}(e_h\langle v\rangle)-\frac{12\mu}{e_h}\langle v\rangle+\rho g e_h\frac{\partial \acute{Z}}{\partial y}$$

$$\left.-e_h\frac{\partial p}{\partial y}\right] \qquad \text{(C.13)}$$

Equations are integrated numerically over 'u' velocity cells and 'v' velocity cells accordingly and over a small time interval (average velocities are shown without the angle brackets for simplicity from now on).

$$\begin{aligned}
&\frac{(\rho e_h u)_{i,J}^{n+1} - (\rho e_h u)_{i,J}^{n}}{\Delta t} \\
&\quad = -\frac{\left[(e_h \rho u u)_{i,J}^{n+1} - (e_h \rho u u)_{i-1,j}^{n+1}\right]}{\Delta x} \\
&\quad - \frac{\left[(e_h \rho u v)_{i,J}^{n+1} - (e_h \rho u v)_{i,J-1}^{n+1}\right]}{\Delta y} \\
&\quad + \mu \frac{\left[(e_h u)_{i-1,J}^{n+1} - 2(e_h u)_{i,J}^{n+1} + (e_h u)_{i+1,J}^{n+1}\right]}{\Delta x^2} \\
&\quad + \mu \frac{\left[(e_h u)_{i,J-1}^{n+1} - 2(e_h u)_{i,J}^{n+1} + (e_h u)_{i,J+1}^{n+1}\right]}{\Delta y^2} - \frac{12\mu}{e_h} u_{i,J}^{n+1} \\
&\quad + (\rho g e_h)_{i,J}^{n} \frac{\Delta z}{\Delta x} - \left(e_h \frac{\Delta p}{\Delta x}\right)_{i,J}^{n+1}
\end{aligned} \tag{C.14}$$

$$\begin{aligned}
&\frac{(\rho e_h v)_{I,j}^{n+1} - (\rho e_h v)_{i,j}^{n}}{\Delta t} \\
&\quad = -\frac{\left[(e_h \rho u v)_{I,j}^{n+1} - (e_h \rho u v)_{I-1,j}^{n+1}\right]}{\Delta x} \\
&\quad - \frac{\left[(e_h \rho v v)_{I,j}^{n+1} - (e_h \rho v v)_{I,j-1}^{n+1}\right]}{\Delta y} \\
&\quad + \mu \frac{\left[(e_h v)_{I-1,j}^{n+1} - 2(e_h v)_{I,j}^{n+1} + (e_h v)_{I+1,j}^{n+1}\right]}{\Delta x^2} \\
&\quad + \mu \frac{\left[(e_h v)_{I,j-1}^{n+1} - 2(e_h v)_{I,j}^{n+1} + (e_h v)_{I,j+1}^{n+1}\right]}{\Delta y^2} - \frac{12\mu}{e_h} v_{I,j}^{n+1} \\
&\quad + (\rho g e_h)_{I,j}^{n} \frac{\Delta z}{\Delta y} - \left(e_h \frac{\Delta p}{\Delta y}\right)_{I,j}^{n+1}
\end{aligned} \tag{C.15}$$

C.4.1.2 Continuity equation

$$\int_t^{t+\Delta t}\int_{cv}\left(\frac{\partial}{\partial \mathrm{x}}(\rho e_h u)+\frac{\partial}{\partial \mathrm{x}}(\rho e_h v)\right)dvdt=0 \tag{C.16}$$

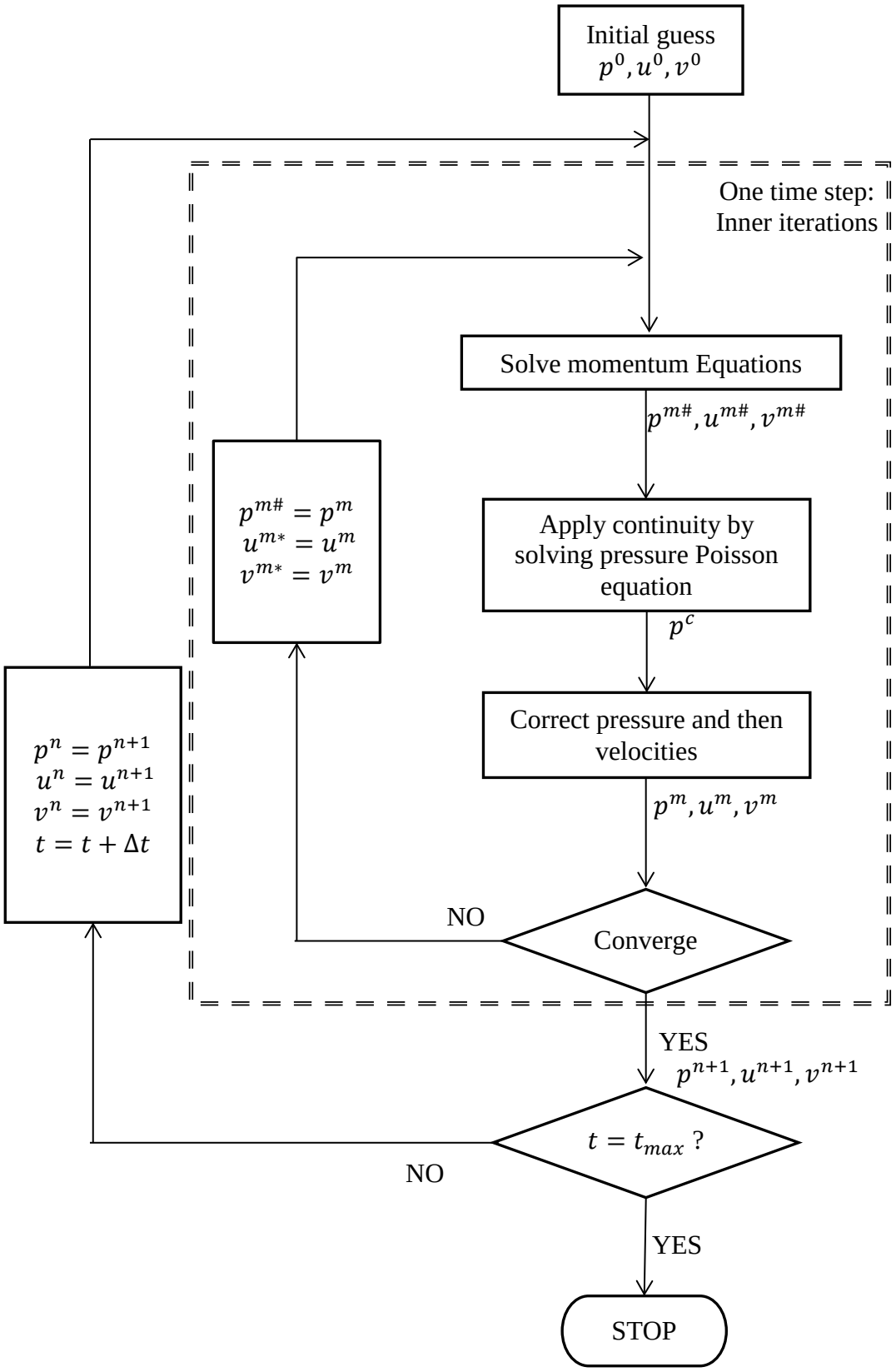
The descretised continuity equation after numerically integrated the equation (C.16) over a scalar cell and over a small time interval is given below.

$$\frac{\left[(\rho e_h u)_{i,J}^{n+1}-(\rho e_h u)_{i-1,J}^{n+1}\right]}{\Delta x}+\frac{\left[(\rho e_h v)_{I,j}^{n+1}-(\rho e_h v)_{I,j-1}^{n+1}\right]}{\Delta y}=0 \tag{C.17}$$

C.4.2 SIMPLE transient algorithm

SIMPLE or SIMPLEC algorithms are used to solve pressure and velocities inside a time step and once a stable solution is sought, the time step is advanced. For the inner iterations in transient calculations, the equations (C.14) and (C.15) are rearranged with explicit terms in RHS. Inside the inner iteration loop, the terms in time interval $n+1$ are shown with superscript 'm' to show that the values are still being updated to reach $n+1$ time interval. Superscript '#' is used as earlier to show the intermediate velocities and superscript '*' for explicit terms.

$$\begin{aligned}
&\frac{(\rho e_h u)_{i,J}^{m\#}}{\Delta t}-\frac{(\mu e_h u)_{i-1,J}^{m\#}}{\Delta x^2}+\frac{2(\mu e_h u)_{i,J}^{m\#}}{\Delta x^2}-\frac{(\mu e_h u)_{i+1,J}^{m\#}}{\Delta x^2}-\frac{(\mu e_h u)_{i,J-1}^{m\#}}{\Delta y^2} \\
&\quad+\frac{2(\mu e_h u)_{i,J}^{m\#}}{\Delta y^2}-\frac{(\mu e_h u)_{i,J+1}^{m\#}}{\Delta y^2}+\frac{12\mu}{e_h}u_{i,J}^{m\#} \\
&\quad=\frac{(\rho e_h u)_{i,J}^{n}}{\Delta t}-\frac{\left[(e_h\rho uu)_{i,J}^{m*}-(e_h\rho uu)_{i-1,j}^{m*}\right]}{\Delta x} \\
&\quad-\frac{\left[(e_h\rho uv)_{i,J}^{m*}-(e_h\rho uv)_{i,J-1}^{m*}\right]}{\Delta y}+(\rho g e_h)_{i,J}^{n}\frac{\Delta z}{\Delta x} \\
&\quad-\left(e_h\frac{\Delta p}{\Delta x}\right)_{i,J}^{m\#}=\phi_{i,J}
\end{aligned} \tag{C.18}$$


Initial guess
p^0, u^0, v^0
One time step:
Inner iterations
Solve momentum Equations
$p^{m\#}, u^{m\#}, v^{m\#}$
$p^{m\#} = p^m$
$u^{m*} = u^m$
$v^{m*} = v^m$
Apply continuity by solving pressure Poisson equation
p^c
Correct pressure and then velocities
p^m, u^m, v^m
NO
Converge
YES
$p^{n+1}, u^{n+1}, v^{n+1}$
$t = t_{max}$?
NO
YES
STOP
$p^n = p^{n+1}$
$u^n = u^{n+1}$
$v^n = v^{n+1}$
$t = t + \Delta t$

$$\frac{(\rho e_h v)_{I,j}^{m\#}}{\Delta t} - \frac{(\mu e_h v)_{I-1,j}^{m\#}}{\Delta x^2} + \frac{2(\mu e_h v)_{I,j}^{m\#}}{\Delta x^2} - \frac{(\mu e_h v)_{I+1,j}^{m\#}}{\Delta x^2} - \frac{(\mu e_h v)_{I,j-1}^{m\#}}{\Delta y^2} \tag{C.19}$$

$$+ \frac{2(\mu e_h v)_{I,j}^{m\#}}{\Delta y^2} - \frac{(\mu e_h v)_{I,j+1}^{m\#}}{\Delta y^2} + \frac{12\mu}{e_h} v_{I,j}^{m\#}$$

$$= \frac{(\rho e_h v)_{i,j}^{n}}{\Delta t} - \frac{\left[(e_h \rho u v)_{I,j}^{m*} - (e_h \rho u v)_{I-1,j}^{m*}\right]}{\Delta x}$$

$$- \frac{\left[(e_h \rho v v)_{I,j}^{m*} - (e_h \rho v v)_{I,j-1}^{m*}\right]}{\Delta y} + (\rho g e_h)_{I,j}^{n} \frac{\Delta z}{\Delta y}$$

$$- \left(e_h \frac{\Delta p}{\Delta y}\right)_{I,j}^{m\#} = \phi_{I,j}$$

Alternatively for component u,

$$e_{h_{(i,J)}} \left[\frac{\rho}{\Delta t} + \frac{2\mu}{\Delta x^2} + \frac{2\mu}{\Delta y^2} + \frac{12\mu}{{e_h}^2}\right] u_{i,J}^{m\#} - e_{h_{(i-1,J)}} \frac{\mu}{\Delta x^2} u_{i-1,J}^{m\#} \tag{C.20}$$

$$- e_{h_{(i+1,J)}} \frac{\mu}{\Delta x^2} u_{i+1,J}^{m\#} - e_{h_{(i,J-1)}} \frac{\mu}{\Delta y^2} u_{i,J-1}^{m\#}$$

$$- e_{h_{(i,J+1)}} \frac{\mu}{\Delta y^2} u_{i,J+1}^{m\#} = \phi_{i,J}$$

For component v,

$$e_{h_{(I,j)}} \left[\frac{\rho}{\Delta t} + \frac{2\mu}{\Delta x^2} + \frac{2\mu}{\Delta y^2} + \frac{12\mu}{{e_h}^2}\right] v_{I,j}^{m\#} - e_{h_{(I-1,j)}} \frac{\mu}{\Delta x^2} v_{I-1,j}^{m\#} \tag{C.21}$$

$$- e_{h_{(I+1,j)}} \frac{\mu}{\Delta x^2} v_{I+1,j}^{m\#} - e_{h_{(I,j-1)}} \frac{\mu}{\Delta y^2} v_{I,j-1}^{m\#}$$

$$- e_{h_{(I,j+1)}} \frac{\mu}{\Delta y^2} v_{I,j+1}^{m\#} = \phi_{I,j}$$

Inner iterations start with solving equations (C.20) and (C.21) for intermediate velocities. Next the same procedure is followed as in SIMPLE/SIMPLEC method described above to correct velocities and pressure. The corrections for the velocities will be;

$$u_{i,J}^{n+1} = u_{i,J}^{m\#} + \frac{1}{D_{i,J}} \frac{\left(\acute{p}_{I+1,J} - \acute{p}_{I,J}\right)}{\Delta x} \tag{C.22}$$

$$v_{I,j}^{n+1} = v_{I,j}^{m\#} + \frac{1}{D_{I,j}} \frac{\left(\acute{p}_{I,J+1} - \acute{p}_{I,J}\right)}{\Delta y} \tag{C.23}$$

Where,

$$D_{i,J} = \left[-\frac{\rho}{\Delta t} - \frac{2\mu}{\Delta x^2} - \frac{2\mu}{\Delta y^2} - \frac{12\mu}{{e_h}_{(i,J)}{}^2} \right]$$

$$D_{I,j} = \left[-\frac{\rho}{\Delta t} - \frac{2\mu}{\Delta x^2} - \frac{2\mu}{\Delta y^2} - \frac{12\mu}{{e_h}_{(I,j)}{}^2} \right]$$

Corrected velocities are substituted into the descretised continuity equation to form the pressure Poisson equation.

$$\frac{\left[(\rho e_h u)_{i,J}^{m\#} + \left(\frac{\rho e_h}{D_{i,J}}\right)^{n+1} \frac{\left(\acute{p}_{I+1,J} - \acute{p}_{I,J}\right)}{\Delta x} - (\rho e_h u)_{i-1,J}^{m\#} - \left(\frac{\rho e_h}{D_{i-1,J}}\right)^{n+1} \frac{\left(\acute{p}_{I,J} - \acute{p}_{I-1,J}\right)}{\Delta x} \right]}{\Delta x} + \tag{C.24}$$

$$\frac{\left[(\rho e_h v)_{I,j}^{m\#} + \left(\frac{\rho e_h}{D_{I,j}}\right)^{n+1} \frac{\left(\acute{p}_{I,J+1} - \acute{p}_{I,J}\right)}{\Delta y} - (\rho e_h v)_{I,j-1}^{n+1} - \left(\frac{\rho e_h}{D_{I,j-1}}\right)^{n+1} \frac{\left(\acute{p}_{I,J} - \acute{p}_{I,J-1}\right)}{\Delta y} \right]}{\Delta y} = 0$$

Alternatively,

$$\left(\frac{\rho e_h}{D_{i,J}\Delta x^2}\right)^{n+1} \acute{p}_{I+1,J} - \left(\frac{\rho e_h}{D_{i,J}\Delta x^2}\right)^{n+1} \acute{p}_{I,J} - \left(\frac{\rho e_h}{D_{i-1,J}\Delta x^2}\right)^{n+1} \acute{p}_{I,J} \tag{C.25}$$

$$+ \left(\frac{\rho e_h}{D_{i-1,J}\Delta x^2}\right)^{n+1} \acute{p}_{I-1,J} + \left(\frac{\rho e_h}{D_{I,j}\Delta y^2}\right)^{n+1} \acute{p}_{I,J+1}$$

$$- \left(\frac{\rho e_h}{D_{I,j}\Delta y^2}\right)^{n+1} \acute{p}_{I,J} - \left(\frac{\rho e_h}{D_{I,j-1}\Delta y^2}\right)^{n+1} \acute{p}_{I,J}$$

$$+ \left(\frac{\rho e_h}{D_{I,j-1}\Delta y^2}\right)^{n+1} \acute{p}_{I,J-1}$$

$$= -\frac{(\rho e_h u)_{i,J}^{m\#} - (\rho e_h u)_{i-1,J}^{m\#}}{\Delta x} - \frac{(\rho e_h v)_{I,j}^{m\#} - (\rho e_h v)_{I,j-1}^{n+1}}{\Delta y}$$

Alternatively,

$$H_{I,J}\acute{p}_{I,J} + H_{I+1,J}\acute{p}_{I+1,J} + H_{I-1,J}\acute{p}_{I-1,J} + H_{I,J+1}\acute{p}_{I,J+1} + H_{I,J-1}\acute{p}_{I,J-1} \tag{C.26}$$

$$= -\frac{(\rho e_h u)_{i,J}^{m\#} - (\rho e_h u)_{i-1,J}^{m\#}}{\Delta x} - \frac{(\rho e_h v)_{I,j}^{m\#} - (\rho e_h v)_{I,j-1}^{n+1}}{\Delta y}$$

Where,

$$H_{I+1,J} = \left(\frac{\rho e_h}{D_{i,J}\Delta x^2}\right)^{n+1} \acute{p}_{I+1,J}$$

$$H_{I-1,J} = \left(\frac{\rho e_h}{D_{i-1,J}\Delta x^2}\right)^{n+1}$$

$$H_{I,J+1} = \left(\frac{\rho e_h}{D_{I,j}\Delta y^2}\right)^{n+1}$$

$$H_{I,J-1} = \left(\frac{\rho e_h}{D_{I,j-1}\Delta y^2}\right)^{n+1}$$

$$H_{I-1,J} = -H_{I-1,J} - H_{I+1,J} - H_{I,J-1} - H_{I,J+1}$$

C.4.3 *SIMPLEC Transient Algorithm*

For transient SIMPLEC method too, the equations (C.20) and (C.21) are used to calculate the intermediate velocities. Once the intermediate velocities are calculated in a particular time step, then the velocity and pressure correction equations are derived. In SIMPLEC the neighbour cell velocity corrections are assumed to be equal to the middle cell velocity corrections. Then the velocity and pressure corrections are related as follows;

$$\acute{u}_{i,J} = \frac{e_{h(i,J)}}{D_{i.J}} \frac{\left(\acute{p}_{I+1,J} - \acute{p}_{I,J}\right)}{\Delta x} \tag{C.27}$$

$$\acute{v}_{i,J} = \frac{e_{h(I,j)}}{D_{I,J}} \frac{\left(\acute{p}_{I,J+1} - \acute{p}_{I,J}\right)}{\Delta y} \tag{C.28}$$

Where,

$$D_{i.J} = -\frac{\rho}{\Delta t} - \frac{12\mu}{e_{h(i,J)}} + \left(e_{h(i-1,J)} + e_{h(i+1,J)} - 2e_{h(i,J)}\right)\left[\frac{\mu}{\Delta x^2}\right] + \left(e_{h(i,J-1)} + e_{h(i,J+1)} - 2e_{h(i,J)}\right)\left[\frac{\mu}{\Delta y^2}\right] \tag{C.29}$$

$$D_{I,j} = -\frac{\rho}{\Delta t} - \frac{12\mu}{e_{h(I,j)}} + \left(e_{h(I-1,j)} + e_{h(I+1,j)} - 2e_{h(I,j)}\right)\left[\frac{\mu}{\Delta x^2}\right] + \left(e_{h(I,j-1)} + e_{h(I,j+1)} - 2e_{h(I,j)}\right)\left[\frac{\mu}{\Delta y^2}\right] \tag{C.30}$$

Pressure correction equations are same as SIMPLE transient method except the expressions $D_{i.J}$ and $D_{I,j}$ which should be calculated using the equations (C.29) and (C.30) respectively.

C.5 Flow with Material Transport

Flow with material transport can be modelled as multi-phase flow. A volume fraction is defined for each phase so that the summation of all the volume fractions becomes

unity. Each phase gets a momentum equation and continuity equation for the flow. All of them are descretised and solved using SIMPLE algorithm. All the governing equations are solved in every iteration cycle. Pressure is assumed to be shared by both phases and iterate until all the momentum and continuity equations are satisfied with a single pressure distribution. In the laboratory tri-axial tests, clean water will be the input flow and the erosions that take place inside the fracture will be the source of sediments. This source appears in the continuity equation of sediments. Since the effective densities are used in governing equations for phases, the density changes with time. Then the averaged continuity equations will have additional term to account the density changes with time.

C.5.1 *Sediment Phase Momentum Equation*

The momentum equations are integrated over a control volume and a small time step.

$$\int_{t}^{t+\Delta t}\int_{cv}\left[\frac{\partial(C_s\rho_s e_h\langle \boldsymbol{V}\rangle_s)}{\partial t}+\acute{\boldsymbol{\nabla}}.(C_s\rho_s e_h\langle \boldsymbol{V}\rangle_s\langle \boldsymbol{V}\rangle_s)+C_s e_h\acute{\boldsymbol{\nabla}}p-C_s\rho_s e_h g\acute{\boldsymbol{\nabla}}Z\right.$$
$$\left.-C_s\mu\acute{\nabla}^2(e_h\langle \boldsymbol{V}\rangle_s)+\frac{12C_s\mu}{e_h}\langle \boldsymbol{V}\rangle_s\right]dvdt=0 \tag{C.31}$$

The scalar form of the momentum equation is taken and integrated as same as in transient flow section to obtain following equations (C.32) and (C.33). Then they are re-arranged in order to solve for the intermediate velocities of SIMPLE/SIMPLEC algorithms.

$$\frac{(C_s\rho_s e_h u)_{i,J}^{m\#}}{\Delta t} - \frac{(C_s\mu e_h u)_{i-1,J}^{m\#}}{\Delta x^2} + \frac{2(C_s\mu e_h u)_{i,J}^{m\#}}{\Delta x^2} - \frac{(C_s\mu e_h u)_{i+1,J}^{m\#}}{\Delta x^2} \tag{C.32}$$

$$- \frac{(C_s\mu e_h u)_{i,J-1}^{m\#}}{\Delta y^2} + \frac{2(C_s\mu e_h u)_{i,J}^{m\#}}{\Delta y^2} - \frac{(C_s\mu e_h u)_{i,J+1}^{m\#}}{\Delta y^2}$$

$$+ \frac{12C_s\mu}{e_h} u_{i,J}^{m\#}$$

$$= \frac{(C_s\rho_s e_h u)_{i,J}^{n}}{\Delta t} - \frac{\left[(e_h C_s\rho_s uu)_{i,J}^{m*} - (e_h C_s\rho_s uu)_{i-1,j}^{m*}\right]}{\Delta x}$$

$$- \frac{\left[(e_h C_s\rho_s uv)_{i,J}^{m*} - (e_h C_s\rho_s uv)_{i,J-1}^{m*}\right]}{\Delta y} + (C_s\rho_s g e_h)_{i,J}^{n} \frac{\Delta z}{\Delta x}$$

$$- \left(C_s e_h \frac{\Delta p}{\Delta x}\right)_{i,J}^{m\#} = \phi_{i,J}$$

$$\frac{(C_s\rho_s e_h v)_{I,j}^{m\#}}{\Delta t} - \frac{(C_s\mu e_h v)_{I-1,j}^{m\#}}{\Delta x^2} + \frac{2(C_s\mu e_h v)_{I,j}^{m\#}}{\Delta x^2} - \frac{(C_s\mu e_h v)_{I+1,j}^{m\#}}{\Delta x^2} \tag{C.33}$$

$$- \frac{(C_s\mu e_h v)_{I,j-1}^{m\#}}{\Delta y^2} + \frac{2(C_s\mu e_h v)_{I,j}^{m\#}}{\Delta y^2} - \frac{(C_s\mu e_h v)_{I,j+1}^{m\#}}{\Delta y^2}$$

$$+ \frac{12C_s\mu}{e_h} v_{I,j}^{m\#}$$

$$= \frac{(C_s\rho_s e_h v)_{i,j}^{n}}{\Delta t} - \frac{\left[(e_h C_s\rho_s uv)_{I,j}^{m*} - (e_h C_s\rho_s uv)_{I-1,j}^{m*}\right]}{\Delta x}$$

$$- \frac{\left[(e_h C_s\rho_s vv)_{I,j}^{m*} - (e_h C_s\rho_s vv)_{I,j-1}^{m*}\right]}{\Delta y} + (C_s\rho_s g e_h)_{I,j}^{n} \frac{\Delta z}{\Delta y}$$

$$- \left(C_s e_h \frac{\Delta p}{\Delta y}\right)_{I,j}^{m\#} = \phi_{I,j}$$

Alternatively,

$$(C_s e_h)_{(i,J)}\left[\frac{\rho_s}{\Delta t}+\frac{2\mu}{\Delta x^2}+\frac{2\mu}{\Delta y^2}+\frac{12\mu}{{e_h}^2}\right]u_{i,J}^{m\#}-(C_s e_h)_{(i-1,J)}\frac{\mu}{\Delta x^2}u_{i-1,J}^{m\#} \tag{C.34}$$

$$-(C_s e_h)_{(i+1,J)}\frac{\mu}{\Delta x^2}u_{i+1,J}^{m\#}-(C_s e_h)_{(i,J-1)}\frac{\mu}{\Delta y^2}u_{i,J-1}^{m\#}$$

$$-(C_s e_h)_{(i,J+1)}\frac{\mu}{\Delta y^2}u_{i,J+1}^{m\#}=\phi_{i,J}$$

$$(C_s e_h)_{(I,j)}\left[\frac{\rho_s}{\Delta t}+\frac{2\mu}{\Delta x^2}+\frac{2\mu}{\Delta y^2}+\frac{12\mu}{{e_h}^2}\right]v_{I,j}^{m\#}-(C_s e_h)_{(I-1,j)}\frac{\mu}{\Delta x^2}v_{I-1,j}^{m\#} \tag{C.35}$$

$$-(C_s e_h)_{(I+1,j)}\frac{\mu}{\Delta x^2}v_{I+1,j}^{m\#}-(C_s e_h)_{(I,j-1)}\frac{\mu}{\Delta y^2}v_{I,j-1}^{m\#}$$

$$-(C_s e_h)_{(I,j+1)}\frac{\mu}{\Delta y^2}v_{I,j+1}^{m\#}=\phi_{I,j}$$

When corrections are introduced to velocities and pressure and general SIMPLE assumptions are made, the corrections for velocities are as follows;

$$u_{i,J}^{n+1}=u_{i,J}^{m\#}+\frac{1}{D_{i,J}}\frac{\left(\acute{p}_{I+1,J}-\acute{p}_{I,J}\right)}{\Delta x} \tag{C.36}$$

$$v_{I,j}^{n+1}=v_{I,j}^{m\#}+\frac{1}{D_{I,j}}\frac{\left(\acute{p}_{I,J+1}-\acute{p}_{I,J}\right)}{\Delta y} \tag{C.37}$$

Where,

$$D_{i,J}=\left[-\frac{\rho_s}{\Delta t}-\frac{2\mu}{\Delta x^2}-\frac{2\mu}{\Delta y^2}-\frac{12\mu}{{e_{h_{(i,J)}}}^2}\right]$$

$$D_{I,j}=\left[-\frac{\rho_s}{\Delta t}-\frac{2\mu}{\Delta x^2}-\frac{2\mu}{\Delta y^2}-\frac{12\mu}{{e_{h_{(I,j)}}}^2}\right]$$

C.5.2 *Sediment Continuity Equation*

The continuity equation for the sediment phase is integrated over a control volume and time interval as earlier;

$$\int_{t}^{t+\Delta t}\int_{cv}\left(e_h\frac{\partial(C_s\rho_s)}{\partial t}+\frac{\partial}{\partial \mathrm{x}}(C_s\rho_s e_h u)+\frac{\partial}{\partial \mathrm{x}}(C_s\rho_s e_h v)-r_e\right)dvdt=0 \tag{C.38}$$

$$(e_h)_{I,J}^{n+1}\frac{\left[(C_s\rho_s)_{I,J}^{n+1}-(C_s\rho_s)_{I,J}^{n}\right]}{\Delta t}+\frac{\left[(C_s\rho_s e_h u)_{i,J}^{n+1}-(C_s\rho_s e_h u)_{i-1,J}^{n+1}\right]}{\Delta x} \tag{C.39}$$

$$+\frac{\left[(C_s\rho_s e_h v)_{I,j}^{n+1}-(C_s\rho_s e_h v)_{I,j-1}^{n+1}\right]}{\Delta y}-(r_e)_{I,J}^{n+1}=0$$

Now corrected velocities (C.36) and (C.37) are substituted into the equation (C.39) accordingly.

$$\frac{\left[\left\{(C_s\rho_s e_h u)_{i,J}^{m\#}+\frac{(C_s\rho_s e_h)_{i,J}}{D_{i,J}}^{n+1}\frac{(\dot{p}_{I+1,J}-\dot{p}_{I,J})}{\Delta x}\right\}-\left\{(C_s\rho_s e_h u)_{i-1,J}^{m\#}+\frac{(C_s\rho_s e_h)_{i-1,J}}{D_{i-1,J}}^{n+1}\frac{(\dot{p}_{I,J}-\dot{p}_{I-1,J})}{\Delta x}\right\}\right]}{\Delta x}+ \tag{C.40}$$

$$\frac{\left[\left\{(C_s\rho_s e_h v)_{I,j}^{m\#}+\frac{(C_s\rho_s e_h)_{I,j}}{D_{I,j}}^{n+1}\frac{(\dot{p}_{I,J+1}-\dot{p}_{I,J})}{\Delta y}\right\}-\left\{(C_s\rho_s e_h v)_{I,j-1}^{m\#}+\frac{(C_s\rho_s e_h)_{I,j-1}}{D_{I,j-1}}^{n+1}\frac{(\dot{p}_{I,J}-\dot{p}_{I,J-1})}{\Delta y}\right\}\right]}{\Delta y}=$$

$$(r_e)_{I,J}^{n+1}-(e_h)_{I,J}^{n+1}\frac{\left[(C_s\rho_s)_{I,J}^{n+1}-(C_s\rho_s)_{I,J}^{n}\right]}{\Delta t}$$

Alternatively,

$$\frac{(C_s\rho_s e_h)_{i,J}}{D_{i,J}\Delta x^2}^{n+1} \acute{p}_{I+1,J} - \frac{(C_s\rho_s e_h)_{i,J}}{D_{i,J}\Delta x^2}^{n+1} \acute{p}_{I,J} - \frac{(C_s\rho_s e_h)_{i-1,J}}{D_{i-1,J}\Delta x^2}^{n+1} \acute{p}_{I,J} \tag{C.41}$$

$$+ \frac{(C_s\rho_s e_h)_{i-1,J}}{D_{i-1,J}\Delta x^2}^{n+1} \acute{p}_{I-1,J} + \frac{(C_s\rho_s e_h)_{I,j}}{D_{I,j}\Delta y^2}^{n+1} \acute{p}_{I,J+1}$$

$$- \frac{(C_s\rho_s e_h)_{I,j}}{D_{I,j}\Delta y^2}^{n+1} \acute{p}_{I,J} - \frac{(C_s\rho_s e_h)_{I,j-1}}{D_{I,j-1}\Delta y^2}^{n+1} \acute{p}_{I,J}$$

$$+ \frac{(C_s\rho_s e_h)_{I,j-1}}{D_{I,j-1}\Delta y^2}^{n+1} \acute{p}_{I,J-1}$$

$$= -\frac{(C_s\rho_s e_h u)_{i,J}^{m\#} - (C_s\rho_s e_h u)_{i-1,J}^{m\#}}{\Delta x}$$

$$- \frac{(C_s\rho_s e_h v)_{I,j}^{m\#} - (C_s\rho_s e_h v)_{I,j-1}^{m\#}}{\Delta y} + (r_e)_{I,J}^{n+1}$$

$$- (e_h)_{I,J}^{n+1} \frac{\left[(C_s\rho_s)_{I,J}^{n+1} - (C_s\rho_s)_{I,J}^{n}\right]}{\Delta t}$$

Alternatively,

$$E_{I,J}\acute{p}_{I,J} + E_{I+1,J}\acute{p}_{I+1,J} + E_{I-1,J}\acute{p}_{I-1,J} + E_{I,J+1}\acute{p}_{I,J+1} + E_{I,J-1}\acute{p}_{I,J-1} = \Phi_{I,J} \tag{C.42}$$

Where,

$$E_{I+1,J} = \frac{(C_s\rho_s e_h)_{i,J}}{D_{i,J}\Delta x^2}^{n+1}$$

$$E_{I-1,J} = \frac{(C_s\rho_s e_h)_{i-1,J}}{D_{i-1,J}\Delta x^2}^{n+1}$$

$$E_{I,J+1} = \frac{(C_s\rho_s e_h)_{I,j}}{D_{I,j}\Delta y^2}^{n+1}$$

$$E_{I,J-1} = \frac{(C_s\rho_s e_h)_{I,j-1}}{D_{I,j-1}\Delta y^2}^{n+1}$$

$$E_{I,J} = -E_{I+1,J} - E_{I-1,J} - E_{I,J+1} - E_{I,J-1}$$

$$\Phi_{I,J} = -\frac{(C_s\rho_s e_h u)^{m\#}_{i,J} - (C_s\rho_s e_h u)^{m\#}_{i-1,J}}{\Delta x} - \frac{(C_s\rho_s e_h v)^{m\#}_{I,j} - (C_s\rho_s e_h v)^{m\#}_{I,j-1}}{\Delta y}$$

$$+ (r_e)^{n+1}_{I,J} - (e_h)^{n+1}_{I,J}\frac{\left[(C_s\rho_s)^{n+1}_{I,J} - (C_s\rho_s)^{n}_{I,J}\right]}{\Delta t}$$

Value of $\Phi_{I,J}$ can be determined at the end of the momentum equation solution for intermediate velocities. Solution method for the fluid phase too is the same.

C.5.3 *Momentum Equation for the Fluid Phase*

The descretised momentum equations for the fluid phase can be obtained as follows;

$$([1 - C_s]e_h)_{(i,J)}\left[\frac{\rho_f}{\Delta t} + \frac{2\mu}{\Delta x^2} + \frac{2\mu}{\Delta y^2} + \frac{12\mu}{{e_h}^2}\right]u^{m\#}_{i,J} \tag{C.43}$$

$$- ([1 - C_s]e_h)_{(i-1,J)}\frac{\mu}{\Delta x^2}u^{m\#}_{i-1,J}$$

$$- ([1 - C_s]e_h)_{(i+1,J)}\frac{\mu}{\Delta x^2}u^{m\#}_{i+1,J}$$

$$- ([1 - C_s]e_h)_{(i,J-1)}\frac{\mu}{\Delta y^2}u^{m\#}_{i,J-1}$$

$$- ([1 - C_s]e_h)_{(i,J+1)}\frac{\mu}{\Delta y^2}u^{m\#}_{i,J+1}$$

$$= \frac{\left([1 - C_s]\rho_f e_h u\right)^{n}_{i,J}}{\Delta t}$$

$$- \frac{\left[\left(e_h[1 - C_s]\rho_f uu\right)^{m*}_{i,J} - \left(e_h[1 - C_s]\rho_f uu\right)^{m*}_{i-1,j}\right]}{\Delta x}$$

$$- \frac{\left[\left(e_h[1 - C_s]\rho_f uv\right)^{m*}_{i,J} - \left(e_h C_s\rho_f uv\right)^{m*}_{i,J-1}\right]}{\Delta y}$$

$$+ \left([1 - C_s]\rho_f g e_h\right)^{n}_{i,J}\frac{\Delta z}{\Delta x} - \left(C_f e_h\frac{\Delta p}{\Delta x}\right)^{m\#}_{i,J} = \phi_{i,J}$$

$$([1-C_s]e_h)_{(I,j)}\left[\frac{\rho_f}{\Delta t}+\frac{2\mu}{\Delta x^2}+\frac{2\mu}{\Delta y^2}+\frac{12\mu}{{e_h}^2}\right]v_{I,j}^{m\#} \qquad \text{(C.44)}$$

$$-([1-C_s]e_h)_{(I-1,j)}\frac{\mu}{\Delta x^2}v_{I-1,j}^{m\#}$$

$$-([1-C_s]e_h)_{(I+1,j)}\frac{\mu}{\Delta x^2}v_{I+1,j}^{m\#}$$

$$-([1-C_s]e_h)_{(I,j-1)}\frac{\mu}{\Delta y^2}v_{I,j-1}^{m\#}$$

$$-([1-C_s]e_h)_{(I,j+1)}\frac{\mu}{\Delta y^2}v_{I,j+1}^{m\#}$$

$$=\frac{\left([1-C_s]\rho_f e_h v\right)_{i,j}^{n}}{\Delta t}$$

$$-\frac{\left[\left(e_h[1-C_s]\rho_f uv\right)_{I,j}^{m*}-\left(e_h[1-C_s]\rho_f uv\right)_{I-1,j}^{m*}\right]}{\Delta x}$$

$$-\frac{\left[\left(e_h[1-C_s]\rho_f vv\right)_{I,j}^{m*}-\left(e_h[1-C_s]\rho_f vv\right)_{I,j-1}^{m*}\right]}{\Delta y}$$

$$+\left([1-C_s]\rho_f g e_h\right)_{I,j}^{n}\frac{\Delta z}{\Delta y}-\left([1-C_s]e_h\frac{\Delta p}{\Delta y}\right)_{I,j}^{m\#}=\phi_{I,j}$$

Pressure correction equation will be;

$$G_{I,J}\acute{p}_{I,J}+G_{I+1,J}\acute{p}_{I+1,J}+G_{I-1,J}\acute{p}_{I-1,J}+G_{I,J+1}\acute{p}_{I,J+1}+G_{I,J-1}\acute{p}_{I,J-1}=\Phi_{I,J} \qquad \text{C.45}$$

Where,

$$G_{I+1,J}=\frac{\left([1-C_s]\rho_f e_h\right)_{i,J}^{n+1}}{D_{i,J}\Delta x^2}$$

$$G_{I-1,J}=\frac{\left([1-C_s]\rho_f e_h\right)_{i-1,J}^{n+1}}{D_{i-1,J}\Delta x^2}$$

$$G_{I,J+1}=\frac{\left([1-C_s]\rho_f e_h\right)_{I,j}^{n+1}}{D_{I,j}\Delta y^2}$$

$$G_{I,J-1} = \frac{\left([1 - C_s]\rho_f e_h\right)_{I,j-1}^{n+1}}{D_{I,j-1}\Delta y^2}$$

$$G_{I,J} = -G_{I+1,J} - G_{I-1,J} - G_{I,J+1} - G_{I,J-1}$$

$$\Phi_{I,J} = -\frac{\left([1 - C_s]\rho_f e_h u\right)_{i,J}^{m\#} - \left([1 - C_s]\rho_f e_h u\right)_{i-1,J}^{m\#}}{\Delta x}$$

$$-\frac{\left([1 - C_s]\rho_f e_h v\right)_{I,j}^{m\#} - \left([1 - C_s]\rho_f e_h v\right)_{I,j-1}^{m\#}}{\Delta y}$$

$$-(e_h)_{I,J}^{n+1}\frac{\left[\left([1 - C_s]\rho_f\right)_{I,J}^{n+1} - \left([1 - C_s]\rho_f\right)_{I,J}^{n}\right]}{\Delta t}$$

Simultaneous solution of all momentum equations and continuity equations by transient SIMPLE algorithm provide the velocities, pressures and sediment erosions over the time and space.

3.4.4. Phase Volume Fraction Calculation

Volume fractions of each phase are changing spatially and temporally. Each time step's velocity and pressure calculations are performed using the current volume fractions. At the end of the time step calculations, the erosion calculations are done and then the volume fractions are updated.